SortingHat: Holisitic Resource Optimization for Heterogeneous Cellular Networks

Steven Hong, Jeffrey Mehlman, and Daniel O'Neill
Stanford University Department of Electrical Engineering
Stanford, California 94305
{ hsiying, jmehlman, dconeill } @ stanford.edu

***Abstract*—While heterogeneous cellular networks offer the potential to improve network performance by augmenting coverage and enhancing traffic capacity, resource allocation is extremely complicated due to the heterogeneous and unplanned nature of base station deployments. In this paper, we present *SortingHat*, a tractable optimization model which leverages the advantages of the different heterogeneous components in the network to achieve 3 main objectives: user-cell pairing, power control, and channel assignment. *SortingHat* achieves this by casting the resource allocation problem in a form which can be solved using iterative optimization algorithms such as the constrained convex-concave procedure (CCCP). While algorithms for user-cell pairing, power control, and channel assignment have been studied extensively in the past, this reformulation enables *SortingHat* to take a holistic approach to resource allocation and jointly optimize all three objectives simultaneously. Numerical results show greater than 30-100% improvement in median user throughput relative to prior works in which the objectives are considered separately, rather than jointly.**

I. Introduction

Driven by the proliferation of smartphones and other mobile computing devices, cellular wireless networks are straining from the rapid explosion of bandwidth-intensive applications. While new protocols like LTE will provide some traffic capacity gains, service providers are increasingly favoring a heterogeneous network architecture [1] characterized by multiple tiers of base-stations with smaller, low-power cells deployed within macro cells to either (1) extend coverage where macro cell coverage is poor, or (2) to increase total traffic capacity in areas with dense data usage. Currently, there are two tiers of low power cells: operator deployed micro cells and user-deployed femto cells. In moving away from the old-school paradigm of carefully planned, homogeneous cellular networks where every cell's capabilities were roughly equal and their geographical placements, frequency bands, and transmit powers were painstakingly chosen, operators are faced with new and challenging multi-tier network management tasks.

The difficulty of managing these networks is rooted in the heterogeneous and chaotic nature of the base-stations. Macro, micro, and femto cells all have constrained resources, capabilities, and environmental settings which can vary along a number of axes:

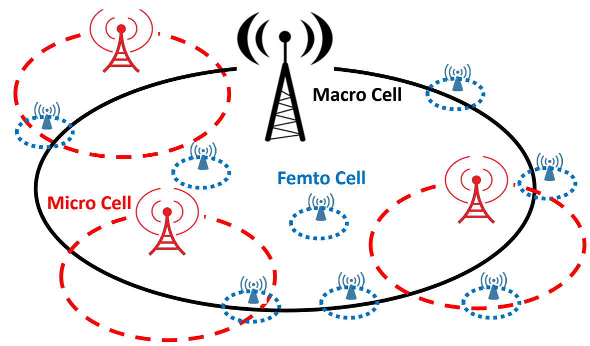

Fig. 1. Heterogeneous Cellular Networks (HCNs) are characterized by the co-existence of different types of cells which vary in their allocatable power, service range, and backhaul capability.

1) **Backhaul Bandwidth:** There exists a high degree of variability in each cell's backhaul capability [2]: most macro cells today have copper T1 links, micro cells have directed microwave links back to aggregation stations, and user-deployed femto cells utilize user-owned broadband connections. With LTE supporting up to 100Mbps in the downlink, cell backhaul capabilities are becoming constraints on a cell's ability to meet traffic demand.
2) **Total Transmit Power/Coverage Range:** While macro cells will operate up to FCC spectral limits, femto and micro cells are limited both by cost limitations and health concerns of the general population, leading to at least an order of magnitude difference in total transmit power capabilities of different cells. As a result of varied transmit power abilities and geographical placement (i.e. on top of a tower, inside a building, etc.), the functional coverage area of every cell can vary from the size of a single home to an entire town. Without careful power control, operators run the risk of causing excessive amounts of interference between cells and choking off the entire network's capacity.
3) **Frequency Range:** While many operators posses narrowband spectral slices which range from 900Mhz to 2GHz, smaller cells may be equipped only to utilize certain portions of the total available spectrum in order to lower deployment cost. As a result, every cell may only be able to utilize a certain number of frequency subchannels.
4) **Local User Density & Demand:** The density of users can vary wildly across time and space (i.e. cities during the day and at night). Additionally, every user can have very different rate demands, ranging from voice (low rate) to streaming video (high rate).

One of the biggest challenges that dense, interference-limited, heterogeneous cellular networks (HCNs) face is in how to effectively allocate resources. In the classical macro-tier setting, users were associated with the cell which provided the best link gain, power control was done using channel inversion, and channel re-use was carefully planned in a static fashion to mitigate interference. However, in HCNs the coverage range of cells across tiers overlap, so it is often not clear how the pairing between users and base-station cells should be made. Furthermore, channel inversion at the macro-tier becomes unusable because of the excessive interference it can cause to lower power tiers and static channel allocation results in over/under provisioning for different cells with varying loads. It is clear that moving forward, *we should jointly address user-cell pairing, power control, and subchannel selection* to find appropriate solutions.

In this work, we develop SortingHat [3], a holistic, multi-tier policy for heterogeneous cellular networks, which builds on prior work on globally optimal, power allocation only schemes [6][7]. In a dense network of cells and users, we unify the choice of user-cell pairings, subchannel selections, and power allocations in the presence of a variety of network constraints. We show that user rate demands can be reformulated into a special *convex-concave* form allowing joint selection of all three parameters. While this reformulation cedes the convexity of the original GP formulation [6], we show that this new, special form allows us to use well understood sequential convex programming methods to find good solutions [20]. We demonstrate that this policy, despite its lack of global convexity, generates sensible, useful results, outperforms existing methods, and provides a holistic framework for analysis of multi-tier heterogeneous networks.

The rest of the paper is organized as follows: In section §II, we highlight some state-of-the-art work in HCN resource allocation. In §III, we describe the problem parameters and variables and develop an optimization program formulation which we will show to be non-convex. In §IV, we present SortingHat, a heuristic solver which utilizes a crisp convex approximation to make the problem tractable. In §V, we simulate and evaluate the performance of the algorithm. In §VI, we detail future directions for this work.

II. Related Work

Industry and academia have proposed a wide variety of resource management algorithms to deal with these multi-tiered complex networks, ranging from very straightforward to extremely complex. These include power-control only algorithms, which assume subchannel and user-cell pairings are made beforehand (simple), to power control and channel selection algorithms which use heuristic approaches like graph coloring (complex), to cell-user pairings which are often considered separately from the other two objectives. Few, if any, of these methods address backhaul limitations of each cell, which are becoming more and more important [2]. We briefly review the state-of-the-art for each of these approaches:

1) ***Power Control:*** If user-cell and subchannel selections are made beforehand, the power control problem can be reformulated into a convex problem [8] for which global solutions are readily found. Proposed solutions include optimal methods based on geometric programming (GP) [6][7] as well as heuristic methods based on power and rate adaptation [4]. Most of these algorithms do not take into account how the transmitters (cells) and receivers (users) are paired and assume this is known a-priori. In this paper, we show that this pairing is crucial, and greatly affects the performance of any power control technique. Because user demands can wildly vary and network resource constraints may dictate that other user-cell pairings are better, separately considering power control, user-cell pairing, and subchannel selection is an inefficient approach and can lead to wasted spectrum. As such, these choices should be made in conjunction with power allocations.

2) ***Power Control & Channel Selection:*** Complex approaches which take some of these joint objectives into consideration have be thoroughly studied [9]-[16], but none cover the entire gambit, and often they are based on empirical observations about network behaviors or unrealistic assumptions. In fact, channel assignment alone for wireless networks has also been studied in the past and has been shown to be NP-hard [10]. A number of heuristics have been proposed for this problem, including centralized, nonlinear convex optimization [11][12], decentralized scheduling problems [16][13][14], and theoretical graph coloring problems [15]. But as with power control, these approaches [10]-[15] assume fixed user-cell pairs, resulting in solutions whose performance is dependent upon the proper user-cell pairings.

3) ***User-Cell Pairing:*** Normally, user-cell pairing relies upon the clustering of certain features such as geographical distance [17], local base station density [18][19], etc. While these approaches can produce good results, they often do not consider multiple constraints (e.g. power, backhaul, interference), and this can lead to significant sub-optimality of their solutions. In general, the entire problem, including power allocation, user-cell association, and subchannel selection, remains NP-hard and thus good heuristics which account for the entire network's characteristics are needed.

III. Problem Formulation and System Model

In this section we develop the system model and motivate the challenges for SortingHat. We consider a wireless downlink network with M cells serving N users. The link gains from cell i to user j, G_{ij}, are represented by the matrix $G \in \mathbf{R}^{M\times N}$. The transmit power from cells i to user j, P_{ij}, are represented by the matrix $P \in \mathbf{R}^{M\times N}$.

Each cell has a maximum total transmit power constraint, specified by the vector $P^{\max} \in \mathbf{R}^{M}$. Additionally, each user demands a certain rate, specified by the vector $r^{\mathbf{des}} \in \mathbf{R}^{N}$, where the network's ability to serve this demand is estimated by a function of SINR. In general, the goal is to choose P such that the aforementioned constraints are satisfied. A cell network may not be able to satisfy every user's demand, so we allow violations of this constraint via a slack rate violation variable, $s \in \mathbf{R}^{N}$, to ensure that the problem is always feasible.

A. Modeling Cell-to-User Rates

We estimate the ability of cell $\hat{i}$ to serve user $\hat{j}$'s rate demand for a particular subchannel $\hat{k}$, $r^{\hat{k}}_{\hat{i}\to\hat{j}}$ via a function of the signal-to-interference-plus-noise (SINR) ratio. We utilize Foschini's curves for an AWGN channel [8], where the available rate (in bits/sec/Hz) for user $\hat{j}$ from cell $\hat{i}$ on subchannel $\hat{k}$ is estimated by

$$r^{\hat{k}}_{\hat{\mathrm{i}}\rightarrow\hat{\mathrm{j}}} \approx \log_2(1 + v\mathrm{SINR}^{\hat{\mathrm{k}}}_{\hat{\mathrm{i}}\rightarrow\hat{\mathrm{j}}}) \tag{1}$$

where v provides rate curves for different modulations. For simplicity, we consider $v = 1$ for this model.

To model the SINR for each user, we consider two interference models representing the dominant multiple access approaches that service providers employ today:

- **Code Division (CDMA):** Each user has a reserved, orthogonal code (subchannel) across all cells, so cells only interfere with each other for a given user and there is no interference between different users. The SINR for a single user-cell pairing in this model is

$$\mathrm{SINR}^{\hat{k}}_{\hat{\mathrm{i}}\rightarrow\hat{\mathrm{j}}} = \frac{G_{\hat{\imath},\hat{\jmath},\hat{k}} P_{\hat{\imath},\hat{\jmath},\hat{k}}}{\sum\limits_{i \neq \hat{\imath}} G_{i,\hat{\jmath},\hat{k}} P_{i,\hat{\jmath},\hat{k}} + \mathcal{N}}$$

  where $\mathcal{N} \in \mathbf{R}$ is the noise power seen by each user and $\hat{k} = \hat{j}$, i.e. each user has it's own channel. We assume that $\mathcal{N}$ is constant across all users.
- **Frequency Division (OFDMA):** Each user utilizes a narrowband subchannel with a cell, and experiences co-channel interference from other users. The SINR for a single user-cell pairing and subchannel in this model is:

$$\mathrm{SINR}^{\hat{k}}_{\hat{\mathrm{i}}\rightarrow\hat{\mathrm{j}}} = \frac{G_{\hat{\imath},\hat{\jmath},\hat{k}} P_{\hat{\imath},\hat{\jmath},\hat{k}}}{\sum\limits_{j \neq \hat{\jmath}} \sum\limits_{i} G_{i,\hat{\jmath},\hat{k}} P_{i,j,\hat{k}} + \sum\limits_{i \neq \hat{\imath}} G_{i,\hat{\jmath},\hat{k}} P_{i,\hat{\jmath},\hat{k}} + \mathcal{N}}$$

  where the first term in the denominator represents the interference from other cells servicing other users on the same subchannel and the second term represents the interference from other cells attempting to service user $\hat{j}$ on the same subchannel.

B. Optimization Problem: Joint BS-user selection, power allocation, and channel assignment

Mathematically, we can express this optimization problem with variables $P \in \mathbf{R}^{M\times N\times K},\ s \in \mathbf{R}^N$ as follows:

$$\begin{array}{ll} \text{minimize} & f(P,s) \quad (2) \\ \text{subject to} & (\textbf{Backhaul bandwidth constraints:}) \\ & \sum\limits_{j=1}^{N} (\log_2(\max\limits_{\hat{i},\hat{k}}(1+\mathrm{SINR}^{\hat{k}}_{\hat{\mathrm{i}}\rightarrow\hat{\mathrm{j}}}))) \leq \beta_i,\ \forall i \quad (3) \\ & (\textbf{Cell transmit power limits:}) \\ & \sum\limits_{k=1}^{K} \sum\limits_{j=1}^{N} P_{i,j,k} \leq P_i^{\max}, \quad i = 1,...,M \quad (4) \\ & (\textbf{Transmit power is nonnegative:}) \\ & P_{i,j,k} \geq 0, \quad \forall\ i,\ j,\ k \quad (5) \\ & (\textbf{Satisfy rate demand minus slack:}) \\ & \log_2(\max\limits_{i,k}(1+\mathrm{SINR}_{\mathrm{i}\rightarrow\mathrm{j}})) \geq r_j^{\mathrm{des}} - s_j,\ \forall j \quad (6) \\ & (\textbf{Slack variable must be nonnegative:}) \\ & s_j \geq 0, \quad j = 1,...,N \quad (7) \end{array}$$

This formulation captures all of the different components of an HCN as discussed in §I. Variation in backhaul capability can be modeled through β_i in constraint (3). The differences in transmit power/coverage range are accounted for by $P_i^{\max}$ in constraint (4). Smaller cells can be restricted to certain subchannels by specifying that $P_{i,j,k} = 0$ for the restricted subchannels k in constraint (5). And lastly, different user rate demands are modeled through varying r_j^{des} in constraint (6).

Depending on the situation, there are several convex objectives which might be of interest such as minimizing total power consumption,

$$f_0(P,s) = \mathbf{1}^T P \mathbf{1} \tag{8}$$

minimizing a weighted sum of user slacks to represent QoS importance on users' demands,

$$f_1(P,s) = w^T s \tag{9}$$

or any weighted combination of these two objective functions

$$f_2(P,s) = \alpha_0 f_0(P,s) + \alpha_1 f_1(P,s) \tag{10}$$

The $\max_{i,k}()$ in constraints (3) and (6) can be motivated as follows: the users' demand should always be served by the best available subchannel and cell SINR, and total user demand on a given cell should not exceed its backhaul ability, respectively. If we pre-allocate a cell i and subchannel k for each user and *ignore the backhaul constraints entirely*, the problem reduces to a convex, geometric program. But pre-allocating user-cell pairs and subchannels precludes the optimization from encompassing all network constraints, which can lead to very suboptimal results. While our formulation unifies user-cell pairing, channel assignment and power allocation, as currently formulated it is not convex. In §IV we address this non-convexity and develop a tractable heuristic.

IV. Algorithm Design

In this section, we present ***SortingHat,*** an HCN resource allocation algorithm which utilizes an iterative, sequential convex programming method, the Constrained Convex-Concave Procedure (CCCP), to find solutions for user subchannel selection, user-cell pairings, and transmit power allocations.

CCCP approximates objectives or constraints which are a difference of two convex functions by taking the negative piece (i.e., the non-convex one), linearizing it locally around a current, feasible point via Taylor approximation, and using this linear approximation to pose a convex function. The approximate convex program is then globally solved, the resulting optimal variable is plugged back into the linearized problem, and this process is repeated until some stopping criterion is met. Because the approximation is local, the algorithm iteratively solves the program, updating the local, Taylor approximations along the way. The effectiveness of the algorithm depends on the specific problem formulation and the chosen starting point, but in many circumstances it works very well. See [20] for more detail on this procedure.

To express our optimization problem as a convex-concave program and use CCCP, we begin by reformulating constraints (3) and (6). Specifically, we re-formulate the $\log_2(\max_{\hat{i},\hat{k}}(1+$

$$
\begin{aligned}
\log_2\left(\max_{\hat{i},\hat{k}}(1+\mathrm{SINR}_{i\to j})\right) &= \log_2\left(\max_{\hat{i},\hat{k}}\left(1+\frac{G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}}}{\sum_{j\neq\hat{j}}\sum_i G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}+\sum_{i\neq\hat{i}}G_{i,\hat{j},\hat{k}}P_{i,\hat{j},\hat{k}}+\mathcal{N}}\right)\right) && (11)\\
&= \log_2\left(\max_{\hat{i},\hat{k}}\left(1+\frac{G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}}}{\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}-G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}}+\mathcal{N}}\right)\right) && (12)\\
&= \log_2\left(\max_{\hat{k}}\left(1+\frac{\max_{\hat{i}}(G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}})}{\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}-\max_{\hat{i}}(G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}})+\mathcal{N}}\right)\right) && (13)\\
&= \max_{\hat{k}}\left(\log_2\left(\frac{\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}+N}{\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}-\max_{\hat{i}}(G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}})+\mathcal{N}}\right)\right) && (14)\\
&= \max_{\hat{k}}\left(\log_2\left(\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}+\mathcal{N}\right)-\log_2\left(\sum_{j,i}G_{i,\hat{j},\hat{k}}P_{i,j,\hat{k}}-\max_{\hat{i}}(G_{i,\hat{j},\hat{k}}P_{i,\hat{j},\hat{k}})+\mathcal{N}\right)\right) && (15)\\
&= \max_{\hat{k}}\Big(\qquad a_{\hat{j}}(P) \qquad - \qquad b_{\hat{j}}(P) \qquad \Big) && (16)
\end{aligned}
$$

$\mathrm{SINR}^{\hat{k}}_{\hat{i}\to\hat{j}}$)) term which appears in both affected constraints as *almost* a fully convex-concave equation, giving both constraints *almost* convex-concave forms. At the top of this page in equations (11)-(16), we step through the algebra required to reform this function and review the reasoning here:

- From eq. (11) to (12), the co-channel interference is rewritten as the summation of the total power emitted by each cell multiplied by the link gains minus the power from the selected cell, $G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}}$.
- From eq. (12) to (13), maximizing the ratio over the i cells can be brought into the numerator and denominator, since the same $G_{\hat{i},\hat{j},\hat{k}}P_{\hat{i},\hat{j},\hat{k}}$ term is affected in both.
- From eq. (13) to (14), $\log()$ is non decreasing on $\mathbf{R}_{++}$ so the maximum over k channels can be pulled out behind the log function.

In its final form (16), the user's optimal rate over all subchannels, cell pairings, and power allocations is a convex-concave function nested inside of a maximization over all k channels.

In order to provide a purely convex-concave form for both of the affected constraints and utilize CCCP, we must first remove the $\max_{\hat{k}}()$ wrapping the function. Thus, within each iteration (iter) of the SortingHat algorithm, we

1) **Select a subchannel $\hat{k}_j$ for each user** j, through a simple, greedy selection scheme in which the users with the largest slack violations are assigned in succession to channels with the minimum co-channel interference. This is shown as steps (3)-(10) in the Pseudo Code.
2) **Optimize with CCCP**, by linearizing the two constraints (rate and backhaul) for the selected subchannel around a current suboptimal power allocation matrix $P^{\text{(iter-1)}}$. With this approximate, convex form in hand, we utilize off-the-shelve solver convex program solver CVX [21]. This is shown as steps (11)-(16) in the Pseudo Code.

After each iteration, power allocation $P^{\text{(iter)}}$ and slack violation $s^{\text{(iter)}}$ are used to seed the next iteration of SortingHat.

SortingHat Pseudo Code:
§IV-A lines (3)-(10); §IV-B lines (11)-(16)

(1) initialize $P^{(0)} \in \mathbf{R}^{M\times N\times K}$ (to feasible starting point)
(2) for iter := 1 to T
(3) Sort($\hat{k}_j, s_j$) pairs by descending values of s_j
(4) for index := 1 to $\min(users \geq K-1)$
(5) $\hat{k}_{index} = argmin_{k\neq\hat{K}}(\sum_{j,i} G_{i,\hat{j},k}P^{(\text{iter}-1)}_{i,j,k} -$
(6) $G_{i,\hat{j},k}P^{(\text{iter}-1)}_{\hat{i},\hat{j},k})$
(7) end
(8) if $(\|obj^{(iter)} - obj^{(iter-1)}\| < 0)$
(9) $P^{(\text{iter})} := P^{(\text{iter}-1)}$
(10) end
(11) Find Taylor expansions around $P^{(iter-1)} = (20), (22)$
(12) $[P^{(\text{iter})}_{\hat{i},\hat{j},\hat{k}}, s_j] =$ CVX.solve($P^{(\text{iter}-1)}_{\hat{i},\hat{j},\hat{k}}$) {
(13) variables: $P^{(\text{iter})} \in \mathbf{R}^{M\times N\times K}, s \in \mathbf{R}^N$
(14) minimize(23)
(15) subject to: $(5), (4), (7), (20), (22)$
(16) };
(17) end

A. Subchannel Selection

Channel sharing among multiple users is a critical issue in mobile cellular systems. In this paper, we focus on channel assignments in OFDMA systems. In this context, the channels we will refer to are equally spaced in the frequency domain and are numerically ordered ($k = 1, 2, 3$ etc.) from low-frequencies to high frequencies. Multiple channels cannot be simultaneously assigned to the same user. To meet the requirements of users with higher rate demands, more power must be allocated to those users on their assigned channel. Aside from this rule, we place no spatial restrictions on channel re-use; adjacent cells are free to re-use the same subchannel if SortingHat decides that is the right choice.

The subchannel selection algorithm works as follows: We first sort the users in descending order by the amount of slack violation they have following the prior iteration of SortingHat. We pick the user with the largest slack and assign it to the channel in which it will experience the smallest amount of co-channel interference (i.e. if no one nearby is using that channel, it's probably a good choice for that user). Then, the user with the 2^{nd} most amount of slack is selected and the procedure is repeated.

The problem with this approach is that oscillations can occur, for instance consider the case when there are N users which are initially assigned to the same channel and thus experience significant co-channel interference. Now assume that there are a total of N available channels, but that $N-1$ of them are already being used by neighboring cells and hence users assigned to these channel will experience a small amount of co-channel interference. If all of the users follow this reassignment algorithm, they will all go to the remaining channel which is free of interference, but in doing so will end up in the situation they were originally in, i.e. all of them interfere with each other. In the next iteration, the channel which they have just vacated will end up being the optimal channel and thus these users will end up oscillating between the 2 channels, and the algorithm would never converge.

Ideally, each subsequent user would be able to account for the interference of the earlier users on new channels before making their new subchannel selection. However, a new power allocation will not be determined until the next iteration of SortingHat, so this is not possible. Our solution for this oscillation problem is to *deny subsequent users from choosing channels which have already been selected by users with more slack in this iteration*. Thus, the maximum number of users that can be assigned to a different channel within each iteration is equal to the total number of channels less one ($K-1$). This approach also has the added benefit that not all users should change, because as some users depart from busy, interference-limited subchannels, these subchannels will have less interference and the performance of the users which have stayed in those subchannels may improve as well.

B. CCCP Formulation

The constraints (3) and (6) of the optimization problem have been reformulated as the difference of two convex problems as shown in eq. (11)-(16) (within a $\max_{\hat{k}}()$ term). Within each iteration of SortingHat, once new subchannel selections for users with the most slack are determined, we can remove the $\max_{\hat{k}}()$ term around both constraints and solve the optimization problem using CCCP [20]. In the final form (16), the first term $a_{\hat{j}}(P)$ is a concave composition and the second term $b_{\hat{j}}(P)$ is the negation of a concave composition (i.e a convex function). Since the rate demand is a concave constraint, we must linearize $b_{\hat{j}}(P)$ in order to utilize CCCP, and vice-versa for the backhaul constraint. We review the linearization procedure for both constraints below:

1) ***Rate Constraint****:* To linearize $b_{\hat{j}}(P)$, we first calculate a subdifferential $\partial b_{\hat{j}}(P)$. Because our space is limited, we show only the subdifferential formulation for the CDMA SINR model, but the calculation is similar for the OFDMA SINR model. To simplify notation, we drop the $\hat{k}$ index (since the channel is already selected) and utilize vector notation, where p_j and g_j are the j^{th} column vectors of P and G, respectively:

$$\partial b_{\hat{j}}(P) = \frac{\left(\sum_{j,i} G_{i,\hat{j}} - \partial\left(\max_{\hat{i}} \mathbf{diag}(g_{\hat{j}})p_{\hat{j}}\right)\right)}{\ln(2)\left(\sum_{j,i} G_{i,\hat{j}}P_{i,j} - \max_{\hat{i}}\left(\mathbf{diag}(g_{\hat{j}})p_{\hat{j}}\right) + \mathcal{N}\right)} \tag{17}$$

The subdifferential of $\max_{\hat{i}}\left(\mathbf{diag}(g_{\hat{j}})p_{\hat{j}}\right)$ is a vector with zeros in all positions except for the i^{th} index of $g_{\hat{j}}$ which provides the maximal $G_{\hat{i},\hat{j}}P_{\hat{i},\hat{j}}$ pair. For example, if the i^{th} index of $\mathbf{diag}(g_j)p_j$ is the maximal entry of the vector, then

$$\partial\left(\max_{\hat{i}} \mathbf{diag}(g_{\hat{j}})p_{\hat{j}}\right) = \left[\begin{array}{ccccccc} 0 & \cdots & 0 & G_{i,\hat{j}} & 0 & \cdots & 0 \end{array}\right]^T \tag{18}$$

Finally, this results in the subdifferential formulation

$$\partial b_{\hat{j}}(P) = \frac{\left[\begin{array}{ccccccc} G_{1,\hat{j}} & \ldots & G_{i-1,\hat{j}} & 0 & G_{i+1,\hat{j}} & \ldots & G_{m,\hat{j}} \end{array}\right]^T}{\ln(2)\left(g_{\hat{j}}^T p_{\hat{j}} - \max_{\hat{i}}\left(\mathbf{diag}(g_{\hat{j}})p_{\hat{j}}\right) + \mathcal{N}\right)} \tag{19}$$

and we have the CCCP approximation for the non-convex rate constraint (6),

$$a_{\hat{j}}(P) - b_{\hat{j}}(P^{(\text{iter}-1)}) - \partial b_{\hat{j}}(P^{(\text{iter}-1)})^T(P - P^{(\text{iter}-1)}) \geq r_{\hat{j}}^{\text{des}} - s_{\hat{j}} \tag{20}$$

where $P^{(\text{iter}-1)}$ is the value of P generated in the last iteration.

2) ***Backhaul Constraint Reformulation****:* A similar approach is taken to satisfy the backhaul constraint (3) but because it is convex, we must linearize the concave component $a_{\hat{j}}(P)$ in order to use CCCP. The subgradient of $a_{\hat{j}}(P)$ is

$$\partial a_{\hat{j}}(P) = \frac{\sum_{j,i} G_{i,\hat{j}}}{\ln(2)\left(\sum_{j,i} G_{i,\hat{j}}P_{i,j} + \mathcal{N}\right)} \tag{21}$$

and the CCCP approximation for the non-concave backhaul constraint(3) becomes

$$\sum_j a_j(P^{(\text{iter}-1)}) - b_j(P) + \partial a_j(P^{(\text{iter}-1)})^T(P - P^{(\text{iter}-1)}) \leq \beta_j \forall j \tag{22}$$

Since the CCCP approximations for equations (20) and (22) are calculated independently and linearize different halves (i.e. $a_{\hat{j}}(P)$ for one and $b_{\hat{j}}(P)$ for the other), the procedure only produces a meaningful result if the opposing linearizations converge to the same result. To ensure this outcome, we augment the objective function such that it minimizes the absolute difference between the two estimates, i.e.

$$\begin{aligned} f_3(P,s) \quad = \quad & \mathbf{1}^T P \mathbf{1} + \alpha \mathbf{1}^T s + \sum_j \mid f_j(P^{(\text{iter}-1)}) \\ & + \partial f_j(P^{(\text{iter}-1)})^T(P - P^{(\text{iter}-1)}) \\ & - b_j(P) - (f_j(P) - b_j(P^{(\text{iter}-1)}) \\ & - \partial b_j(P^{(\text{iter}-1)})^T(P - P^{(\text{iter}-1)}) \mid \end{aligned} \tag{23}$$

Exchanging the original backhaul constraint (3) and original rate constraint (6) for the CCCP constraints (20) and (22) yields a fully convex optimization program which SortingHat can sub-optimally solve using CCCP (since the rest of the problem is convex). This procedure is outlined in steps (11)-(16) of SortingHat's pseudo code.

V. Evaluation

To the best of our knowledge, no current approaches solve the cell-user assignment, power allocation, and channel allocation problem simultaneously. To make fair comparisons with previous work, we analyze the separate components of

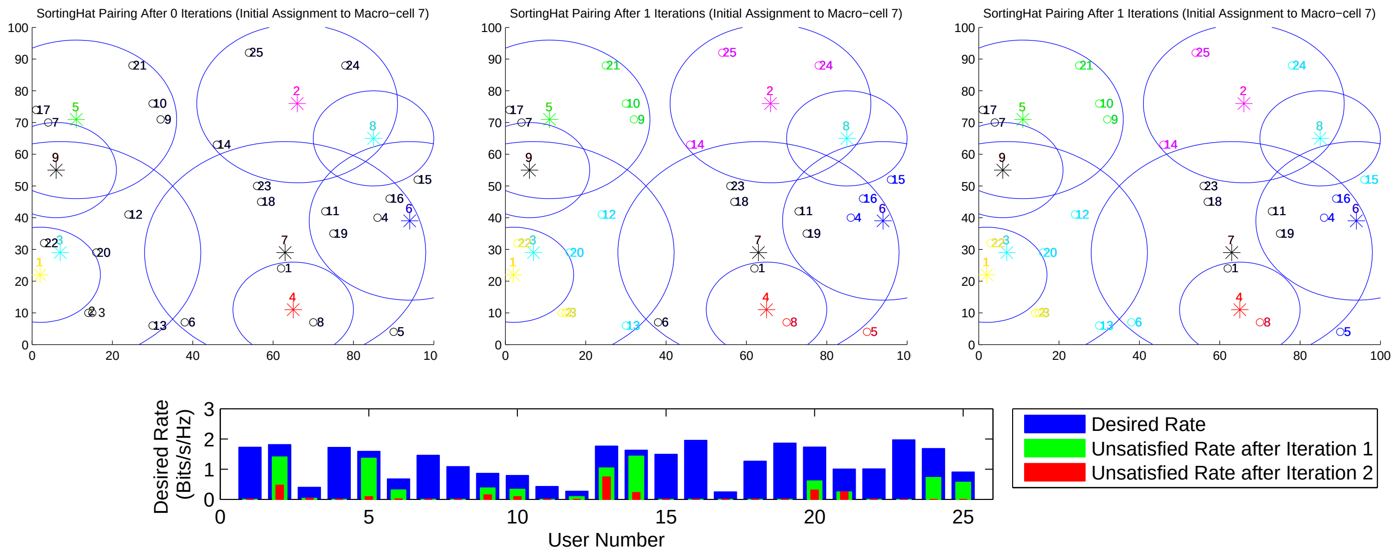

Fig. 2. **Top:** Cell-User Pairing for (A) Initial allocation to Cell 7 (B) After 1 CCCP Iterations, (C) After 2 CCCP Iterations. The colors denote which cell each user is assigned to and the numbers correspond to the user number. **Bottom:** Demanded and Slack Rate per User over 2 CCCP Iterations

SortingHat individually against two prior approaches. First, we compare against an algorithm which performs optimal power allocation using GP [6][7] but uses a clustering approach to assign users to cells first [17]. To obviate the need for channel allocation, we utilize the CDMA interference model for this analysis. Second, we compare our algorithm against an approach [9] where each cell randomly and without replacement assigns a channel to each paired user. Here, we utilize the OFDMA interference model to analyze the performance of SortingHat's channel allocation.

For SortingHat and the compared approaches, the key performance metric in all simulations is the slack violation variable, s, which captures the amount of unsatisfied rate demand per user. For each user, it ranges from $s_j = 0$, i.e. the rate demand is entirely satisfied, to $s_j = r_j^{\text{des}}$, i.e. the rate demand is completely unsatisfied. The goal, of course, is to have $s_j = 0$ for all j.

Test Scenario: We've built a compact simulator to compare the different approaches. To emulate the unplanned deployment patterns of typical HCNs, the simulator randomly place m cells on a two-dimensional square grid of fixed size, $D \times D$. There are n users are scattered throughout the grid, each with a unique rate requirement to mimic different end-user applications. The upper plots in Figure (2) show one such realization with a $D = 100$ service area. The numbered, colored stars denote $m = 9$ cells. There are 3 tiers of cells: Macro, Micro, and Femto. Macro cells (cell 3,7) have total allocatable power of 35 units. Micro cells (cell 2,5,6) have total allocatable power of 25 units and Femto cells (cell 1,4,8,9) can allocate 10 units units of power. All cells are capable of serving every subchannel. The numbered, colored circles denote $n = 25$ users, randomly scattered throughout the service area, and their coloring denotes the user-cell pairing. Using the random positions of the cells and users within the grid, we calculate a corresponding static gain matrix G using basic, line-of-sight RF propagation models. If there are multiple subchannels per cell in the simulation, there is no frequency selectivity (i.e. the gain is flat across subchannels). The blue circles centered around every cell represent the effective service range of each one (based on their total available power).

Finding a Feasible Initialization: The CCCP algorithm requires a feasible starting point for iteration "0". SortingHat is initialized by pre-assigning all users to one particular cell and allocating its total power equally among all of the users. The other cells' powers are initialized to zero.

A. *CDMA Interference Model*

We begin by analyzing the user-cell pairing and power control component of SortingHat under the CDMA Interference Model (where there is one, orthogonal channel per user). The top of Figure 2 shows the progressive user-cell pairings with an initial allocation made to cell 7 over two iterations (Figure2TopA). The bottom of Figure 2 shows the desired rates per user and the evolution of slack (unsatisfied rate) per user over two iterations. After the first iteration many of the users have met their rate demands (i.e., have no slack), while cell-boundary users tend to have significant amounts of slack.

These cell-boundary users are interesting cases because their matchings can dictate the performance of other users. Take user 24 for example, who is serviceable by cell 2 and to a certain extent by cell 8. After the first iteration (Figure 2TopB) it has been assigned to cell 2, but because of the distance between them and the heavy load cell 2 is already supporting, $\sim 50\%$ of user 24's desired rate has not been satisfied. After the second iteration, it is interesting to note that user 24 has switched to cell 8. This switch improves the performance of users 14 and 25 because cell 2 has more power after offloading user 24, while user 24's demand is almost entirely satisfied by cell 8 (Figure 2TopC). Similarly, user 15 also switches to cell 8 despite having all of its rate satisfied in the second iteration. This enables cell 6 to service user 5 and further reduce the total slack rate. By folding the user-cell pairing into the convex optimization constraints, SortingHat is better able to balance the load against the network constraints and improve overall performance.

1. What effect does the initial starting point have? Since any CCCP-based algorithm is not guaranteed to find the global optimum, an important consideration is the effect of the initial

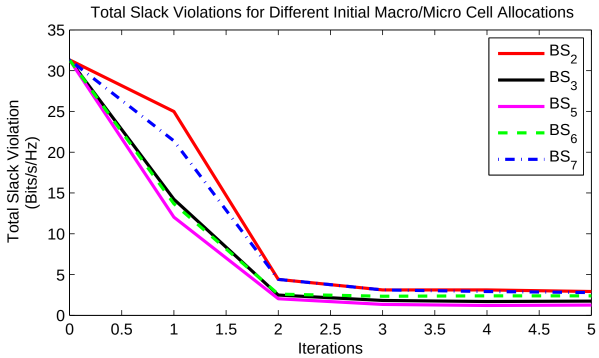

Fig. 3. Total Slack Violation for five initial allocations. All initializations converge to within ±1 Bits/s/Hz of each other

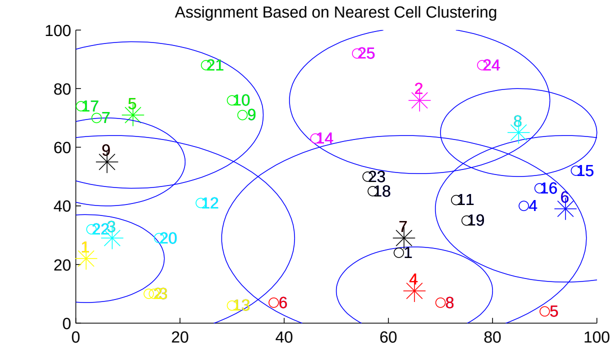

Fig. 5. User-Cell pairs found by nearest cell clustering - Femtos 1 and 4 are oversubscribed while femtos 8 and 9 are under-subscribed.

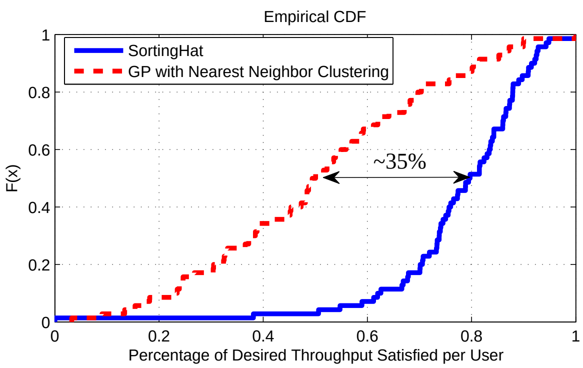

Fig. 4. The median users using SortingHat have almost 80% of their desired throughput satisfied. The same median users using GP with Cell Clustering have less than 45% of their throughput satisfied.

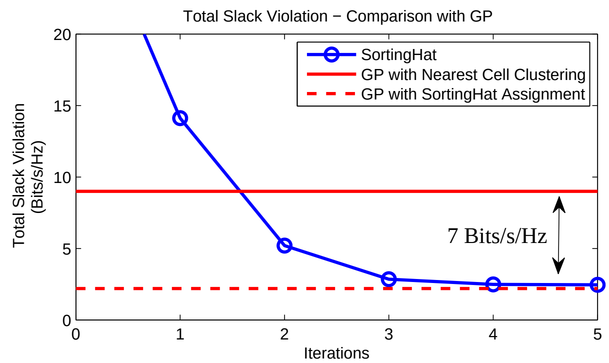

Fig. 6. SortingHat's reduced slack violation corresponds to a 32% increase in network throughput compared to GP with nearest cell clustering and converges to the global optimum found by a GP with SortingHat's user-cell pairing.

starting point. To measure the impact of different starting power allocations, $P^{(0)}$, on the convergence rate and final allocation, Figure 3 shows the total slack violation versus the iteration number for five different initializations. Regardless of starting point, the algorithm converges to the final allocation within 2 to 3 iterations. Interestingly, while some of these initial allocations did provide different P^* final allocations, they all managed to converge to within approximately ± 1 Bits/s/Hz of the same total slack violation objective.

Compared Approach: Power allocation for a predetermined set of user-cell pairings can be re-expressed as a Geometric Program (GP) [6][7] and solved to find globally optimal solutions. While the power allocations are globally optimal for a given set of user-cell pairings, the pairing is fixed and thus does not allow load balancing via cell-user re-pairing. We make the user-cell pairing by clustering users with the cell that provides the maximum link gain $G_{\hat{i},\hat{j}}$. We run the simulator ten times with randomly generated user and cell locations and generate allocations with both approaches.

Figure 4 plots the CDF of the percentage of desired rate satisfied per user, i.e. r_j/r_j^{des}, with the blue line denoting SortingHat's performance and the dotted red line denoting the GP with cell clustering. As you can see, SortingHat significantly outperforms GP with cell clustering because it simultaneously optimizes user-cell pairing and power allocation to load-balance the network. While SortingHat outperforms the GP with cell clustering throughout, the disparity is most obvious in the lower portion of the CDF. With SortingHat, the lower 10% of users have more than 60% of their desired throughput satisfied while with the GP, the same segment of users have less than 25% of their throughput satisfied.

To shed some light on this disparity, Figures 5-7 plot the results for the GP and cell clustering approach, applied to the same scenario as shown in Figure 2. In Figure 5, we can see that assigning users to the nearest cell produces different user-cell pairs than SortingHat did, because name accounts for cell capabilities in its procedure. The cell clustering over-assigns users to femto cells 1 and 4, which were subsequently unable to support the rate demands.

2. How far from optimal is SortingHat? The performance of SortingHat and the GP differs because pairing users to the nearest cell can result in either over provisioning cells, or pairing with suboptimal types (e.g. pairing with low-power femtos which are far away). Total slack violation over five iterations of SortingHat is plotted in Figure 6 and shows the performance gap between SortingHat, a GP with cell clustering, and a GP with the user-cell pairing generated by SortingHat. Interestingly, SortingHat converges to the same global optimum found by the GP using the user-cell pairings generated by SortingHat. Thus for this scenario, *the power-allocation of SortingHat converges to the optimal power allocation for a given set of user-cell pairings.*

3. How do backhaul constraints effect SortingHat? Because nearest cell clustering does not take into account backhaul constraints, SortingHat outperforms GP with clustering even more when backhaul resources are constrained. In Figure 5, the nearest neighbor pairing leaves femto cells 8 and 9 unused, increasing the backhaul and power requirements of cells 2,5, and 6. Because femto cells are not subject to the backhaul constraints of the network, such an allocation is particularly

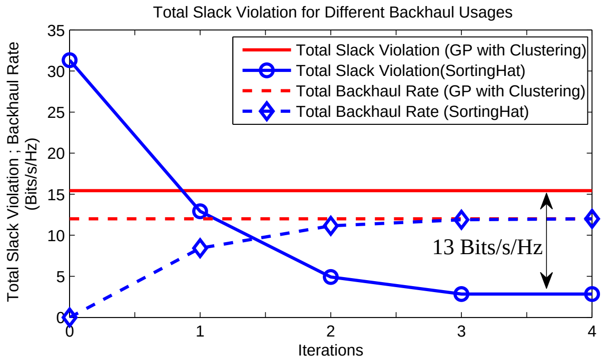

Fig. 7. SortingHat and GP with nearest cell clustering utilize identical backhaul bandwidths but SortingHat reduces slack violations, providing 56% more network throughput by intentionally offloading traffic to femtocells

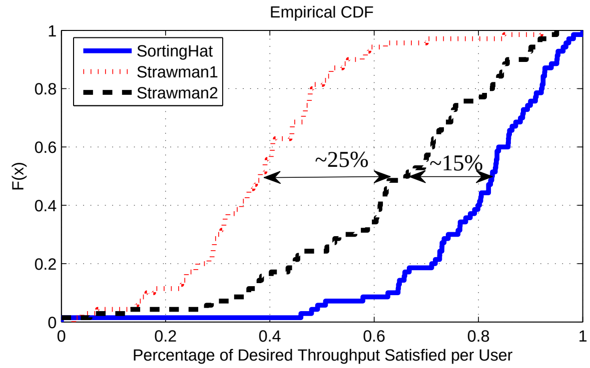

Fig. 8. SortingHat outperforms strawman1 and 2, with the the median users having more than 80% of their desired rate satisfied. The same median segment of users have less than 65% and 40% of their throughput satisfied in strawman1 and 2, respectively.

ineffective when the backhaul is limited. To demonstrate this, Figure 7 shows that when the backhaul resources are finite (represented by the dashed lines), SortingHat utilizes these resources in a much more efficient manner by offloading as much traffic onto the femto cells as possible. As a result, while the backhaul usage is more or less the same (shown by the convergence of the 2 dotted lines in Figure 7, SortingHat provides nearly twice the network throughput that GP can.

B. OFDMA Interference Model - Dynamic Channel Allocation

We now consider an OFDMA model in which simultaneous transmissions between different users interfere with each other if they are assigned to the same channel. Under this interference model, we evaluate SortingHat's subchannel allocation component described in §IV-A. Because the general solution to the channel assignment problem is known to be NP-hard, we compare SortingHat against another heuristic channel allocation algorithm.

Compared Approaches: Given a set of user-cell pairs found with either nearest cell clustering (strawman1) or SortingHat's CCCP approach (strawman2), each cell randomly assigns a channel to each of it's paired users. The cell uniformly selects from the set of channels without replacement until the set is exhausted, and then repeats this process with the original set until all users have been assigned [9]. The total number of channels is the same for SortingHat and both of the strawman approaches. Since the strawman algorithms do not consider inter-cell interference or user-cell pairing, we expect that they will perform poorly in comparison to SortingHat. Again, we run the simulator ten times with randomly generated user and cell locations and generate allocations using all three approaches.

Figure 8 plots the CDF of the percentage of satisfied rate per user, with the blue line denoting the performance of SortingHat, the dotted red line denoting strawman1, and the dot-dash black line denoting strawman2. As expected, SortingHat significantly outperforms both approaches. With SortingHat, the median 50% of users having more than 80% of their desired rate satisfied, while strawman1 and strawman2 satisfy only 65% and 40% of user rate demands, respectively.

Because each approach differs in only a single aspect, we can determine exactly where SortingHat's performance gains come from. First, strawman1 and strawman2 only differ in the user-cell pairing method, and as such the performance advantage can be attributed to SortingHat's iterative pairing which considers important factors like the remaining number of unused channels on each cell. Second, strawman2 and SortingHat differ in only how channels are allocated, thus the performance gap exists because SortingHat considers the interference and distance between different users in its assignment process, whereas strawman2 does not.

To assess these performance results in more depth, Figure 9 shows the channel allocation results for SortingHat, strawman1, and strawman 2 for a particular instance in which there are five channels total. In these figures, the coloring denotes the user-cell pairing, and the numbers next to each user denote which channel it is using. If a user is denoted by an 'x', this means that it's chosen cell has too many users and can't serve that demand.

5. Why does SortingHat outperform strawman1? SortingHat outperforms strawman1 because it takes co-channel interference into account when assigning user-cell pairs and as such, ensures that no cell is assigned more users than the total number of channels available (which in this case is five). In strawman1 in Figure 9B, we see that cell 7 is heavily over-assigned users and as a result is unable to service two of them. Whereas in SortingHat, cell 7 only uses four of the available five channels. This shows SortingHat's ability to take into account important factors, such as the remaining number of unused channels on each cell, when determining user-cell pairs.

6. Why does SortingHat outperform strawman2? SortingHat outperforms strawman2 because it tries to minimize the co-channel interference by assigning repeated channels as far apart as possible. In Figure 8, the performance gap between between SortingHat and strawman2 is much less pronounced than the gap between SortingHat and strawman1 because users are not over assigned to cells. In fact, for low-density usage areas such as the top right corner with cell 2, we found that SortingHat only slightly outperformed strawman2 because the cell's total set of channels is not exhausted. The difference between the channel allocations of SortingHat and strawman2 is subtle. In particular, notice that the user which is closest to cell 7 is actually assigned to cell 4 on channel 1, and this channel is the only one that cell 7 does not use. All of cell 7's neighboring cells, however, *do* use channel 1 to serve users. Since cell 7's service region overlaps almost every other

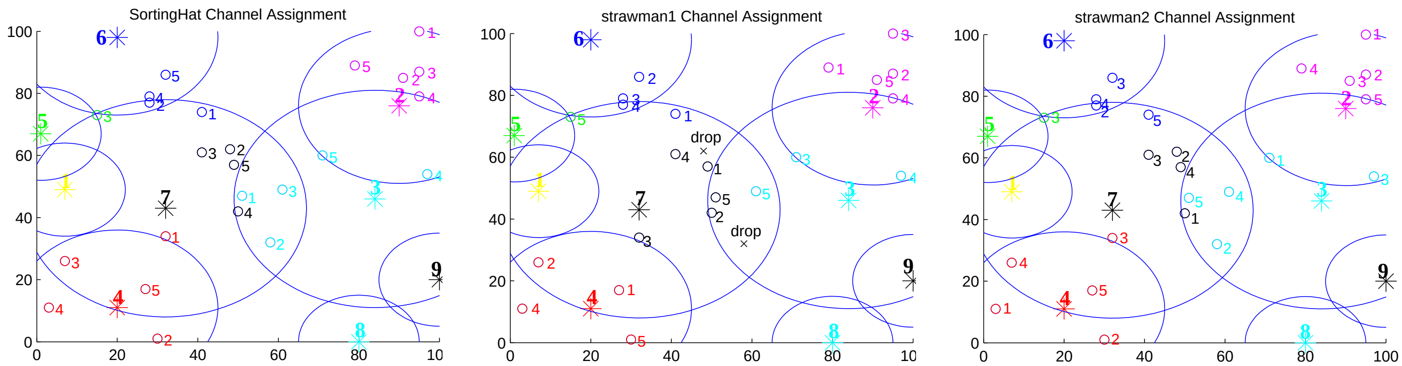

Fig. 9. Final channel allocations for **(A)** SortingHat **(B)** strawman1 **(C)** strawman2. The numbers and colors respectively denote which channel and cell the user has been assigned to.

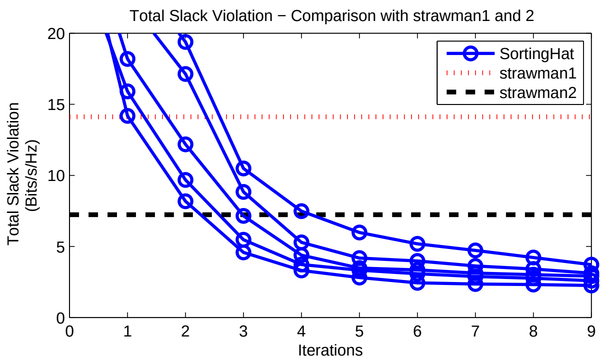

Fig. 10. Total slack violations for different initial cell allocations. All initializations outperform both strawman1 and strawman2 by a margin of 15% to 60% respectively

cell, it creates significant interference on every neighboring cell's channels. This shows SortingHat's natural generation of smart channel re-use; by minimizing the number of users assigned to cell 7, SortingHat is able to minimize the amount of interference caused by cell 7 and effectively re-use the first channel 3 times (cell 6,4,3 all use channel 1) as opposed to once with cell 7.

7. How do different initial starting points affect performance? As we saw in the CDMA model, different initial starting points impact the performance of the CCCP. To measure the impact of the initialization, Figure 10 plots the total slack violation from different starting points in comparison to strawman1 and 2. We see that despite the simplicity of our channel selection algorithm, SortingHat converges to similar solutions, all of which outperform strawman1 and 2.

VI. Conclusion and Future Work

As cellular providers transition from single-tier networks to more heterogeneous deployments, resource allocation will become more challenging. In this paper, we've presented SortingHat, a resource allocation algorithm for HCNs, and show how it can be used to analyze important network performance metrics such as per user throughput, backhaul constraints, and co-channel interference. The benefits of SortingHat were illustrated through several numerical examples, which showed a 30-100% improvement in median user throughput resulting from optimizing user-cell association, power control, and channel assignment jointly rather than separately.

There are numerous possible extensions to this work, which include finding a systematic way to distribute the algorithm to each cell in order to implement SortingHat practically. We also plan to consider real deployments and investigate how an online version of SortingHat adapts to statistically varying gains, mobile users, and changing rate demands.

References

[1] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, Femtocell networks: a survey, In *IEEE Commun. Mag*, 2008.

[2] S. Chia, M. Gasparroni, P. Brick, "The Next Challenge for Cellular Networks: Backhaul," In *IEEE Microwave Mag.*, 2009.

[3] J.K. Rowling, "Harry Potter and the Sorcerer's Stone", *Scholastic Publishing*, 1998.

[4] A. Akella, G. Judd, S. Seshan, P. Steenkiste, "Self-Management in Chaotic Wireless Deployments," In *ACM MOBICOM*, 2005.

[5] V. Chandrasekhar, J. Andrews, T. Muharemovic, Z. Shen, A. Gatherer, "Power Control in Two-Tier Femtocell networks," In *IEEE Trans. W. Comm.*, 2009.

[6] M. Chiang, C. Tan, D. Palomar, D. O'Neill, D. Julian, "Power Control By Geometric Programming," In *IEEE Trans. W. Comm.*, 2007.

[7] D. Julian, M. Chiang, D. O'Neill, S. Boyd, "QOS and Fairness Constrained Convex Optimization of Resource Allocation for Wireless Cellular and Ad Hoc Networks", In *IEEE INFOCOM*, 2002.

[8] G. Foschini, Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," In *IEEE Trans. Veh. Tech.*, 1993.

[9] V. Bhandari, N. Vaidya, "Capacity of Multi-Channel Wireless Networks with Random (c,f) Assignment," In *ACM MOBIHOC*, 2007.

[10] C. Sung, W. Wong, "Sequential Packing Algorithm for Channel Assignment Under Cochannel and Adjacent Channel Interference Constraint," In *IEEE Trans. Veh. Tech.*, 1997.

[11] R. Madan, S. Boyd, S. Lall, "Fast Algorithms for Resource Allocation in Wireless Cellular Networks," In *IEEE Trans. Net.*, 2010.

[12] V. Chandrasekhar, J. Andrews, "Spectrum Allocation in Tiered Cellular Networks," In *IEEE Trans. Comm.*, 2009.

[13] K. Sundaresan, S. Rangarajan, "Efficient Resource Management in OFDMA Femto Cells," In *ACM MOBIHOC*, 2009.

[14] C. Ko, H. Wei, "On-Demand Resource-Sharing Mechanism Design in Two-Tier OFDMA Femtocell Networks," In *IEEE Trans. on Veh. Tech.*, 2011.

[15] K. Ramachandran, E. Belding, K. Almeroth, M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," In *IEEE INFOCOM*, 2006.

[16] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, Y. Wu, "Allocating Dynamic Time-Spectrum Blocks for Cognitive Radio Networks," In *ACM MOBICOM*, 2007.

[17] A. Ganti, T. Klein, M. Haner, "Base Station Assignment and Power Control Algorithms for Data Users in a Wireless Multiaccess Framework," In *IEEE Trans. W. Comm.*, 2006.

[18] R. Yates, "Integrated Power Control and Base Station Assignment," In *IEEE Trans. Veh. Tech.*, 1995.

[19] J. Lee, R. Mazumdar, N. Shroff, "Joint Resource Allocation and Base-Station Assignment for the Downlink in CDMA Networks," In *IEEE/ACM Trans. Net.*, 2006.

[20] B. Sriperumbudur, G. Lanckriet, "On the Convergence of the Concave-Convex Procedure," In *Neural Information Processing Systems*, 2009.

[21] M. Grant, S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming, v. 1.21", *http://cvxr.com/cvx*, 2011.