

Coordinated Cognitive Tethering in Dense Wireless Areas

Haleh Tabrizi, Golnaz Farhadi, John Matthew Cioffi, and Ghadah Aldabbagh

This paper examines the resource gain that can be obtained from the creation of clusters of nodes in densely populated areas. A single node within each such cluster is designated as a “hotspot”; all other nodes then communicate with a destination node, such as a base station, through such hotspots. We propose a semi-distributed algorithm, referred to as coordinated cognitive tethering (CCT), which clusters all nodes and coordinates hotspots to tether over locally available white spaces. CCT performs the following these steps: (a) groups nodes based on a modified k-means clustering algorithm; (b) assigns white-space spectrum to each cluster based on a distributed graph-coloring approach to maximize spectrum reuse, and (c) allocates physical-layer resources to individual users based on local channel information. Unlike small cells (for example, femtocells and WiFi), this approach does not require any additions to existing infrastructure. In addition to providing parallel service to more users than conventional direct communication in cellular networks, simulation results show that CCT can increase the average battery life of devices by 30%, on average.

Keywords: Graph coloring, interference management, k-means clustering, resource allocation, white spaces.

The manuscript received Apr. 20, 2014; revised July 15, 2015; accepted July 22, 2015.

Haleh Tabrizi (corresponding author, htabrizi@stanford.edu) is with the Department of Electrical Engineering, Stanford University, Stanford, CA, USA.

John Matthew Cioffi (cioffi@stanford.edu) is with the Department of Electrical Engineering, Stanford University, Stanford, CA, USA and also with the Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia.

Golnaz Farhadi (gfarhadi@us.fujitsu.com) is with Fujitsu Labs of America, Sunnyvale, CA, USA.

Ghadah Aldabbagh (galdabbagh@kau.edu.sa) is with the Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia.

I. Introduction

With advances in wireless technology and remarkable growth in the number of mobile devices (for example, smartphones and tablets), ubiquitous access to wireless data has emerged as a new demand. This demand is particularly perceived in crowded venues, such as sports stadiums, conference halls, and any other mass gathering of people where a large number of mobile devices contend for limited spectrum. Such crowded venues are filled with people who wish to share their experience with friends through email, SMS, uploading photos to their social networking websites, or simply browsing the web. Venue owners are obliged to provide such communication services through means of a wireless connection to compete with the “in-home” experience that readily provides such amenities through wired or wireless connection.

One traditional approach to solving such an issue in dense wireless areas has been the deployment of additional base stations (BSs); however, this requires cell planning and venue-specific site acquisition [1] and hence increases cost and complexity. Employing advanced physical-layer technologies can increase the spectral efficiency of the limited available spectrum, but it is still not sufficient for today’s massive demands. Khandekar and others have investigated how physical-layer performance is approaching its theoretical limit [2]. Conventional approaches of deploying femtocells or relays, and offloading traffic over WiFi, can improve spectrum congestion, but require fixed deployments. Multiple vendors such as Cisco [3], Ruckus Wireless [4], and Qualcomm [5] have developed solutions to overcome the issue of providing wireless access in densely populated areas, such as in sports stadiums, through the deployment of advanced WiFi networks. Further relocation or addition of infrastructure might be

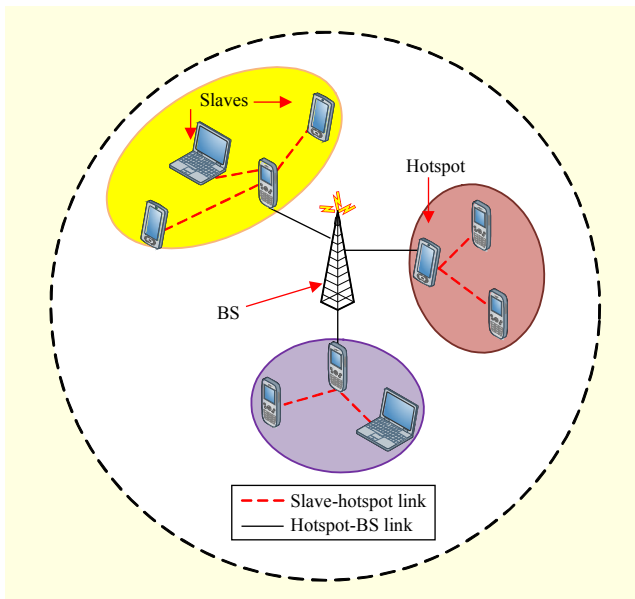


Fig. 1. Proposed hotspot-slave configuration.

necessary based on user distribution in different venues.

We have proposed in [6] the use of some active nodes as “hotspots” to create small cells. Such an approach, in contrast to traditional small cells, does not require additional infrastructure. Furthermore, in comparison to conventional direct mode communication between each node and a BS, the proposed method can increase network capacity by reducing path loss, using locally available white spaces (WSs) within clusters, and reusing spectrum among clusters. This paper extends the proposed method, as well as providing a detailed method of selecting algorithm parameters and a thorough performance evaluation.

Tethering — generally known due to the proliferation of WiFi — is a method of delivering data to a destination node through an intermediate node (hotspot), and it is the core of the proposed coordinated cognitive tethering (CCT) algorithm. A mobile device with cellular access can act as a hotspot to supply broadband access to nearby users, known as *slaves*, as shown in Fig. 1. CCT tries to provide full connectivity by coordinating users, managing interference, and systematically selecting hotspots.

The proposed algorithm systematically exploits unused spectra or WSs for use over the relatively short “hotspot slave links.” There is a growing trend toward a shared spectrum infrastructure, as opposed to exclusive licensed bands. Cognitive radio networks that can autonomously exploit locally unused spectra have been the subject of study for a couple of decades now. A large number of spectrum-sensing algorithms have been developed to cater to the needs of specific communication applications and prevent unauthorized use when primary users are present [7]. Recently, the now

vacated television bands (referred to as TVWS) [8] and shared federal owned spectrum (in the U.S.) [9], [10] have gained much attention.

Relaying over unlicensed bands in an ad hoc manner is not a new idea. Todd and Zhao [11] were among the earliest to investigate this method. However, their focus was on relaying data, which refers to using inactive nodes to deliver data for active nodes. As such, their method employed inactive mobile devices (that is, users were not using their respective mobiles) to deliver data to other users. Furthermore, the method in [11] does not allow more than one slave per hotspot and does not focus on spectrum reuse. CCT, on the other hand, allows each hotspot to assist more than one slave, selects only active nodes to act as hotspots, and tries to maximize spectrum reuse.

This paper investigates whether configuring nodes into hotspots and slaves can improve overall network performance in dense areas. To accomplish this requires intelligently determining a series of unknowns:

- What role (hotspot or slave) should each user take?
- If selected as a slave, to which hotspot should the slave connect?
- On which set of channels should each cluster operate to mitigate interference to other clusters?
- What channels and with how much power should each individual user acquire?

The optimal solution to this problem requires joint optimization of all the problem variables. However, joint optimization requires solving an integer-programming problem that is NP-hard. Hence, with some appropriate assumptions, this paper decomposes the problem into three simpler subproblems.

The proposed algorithm, CCT, consists of clustering nodes into groups, assigning WS spectrum to each cluster, and allocating specific WS or licensed-band channels to individual users based on instantaneous channel gains. This paper’s main contribution is a semi-distributed algorithm that configures nodes into hotspots and slaves and allows the reuse of locally available WS resources. The proposed algorithm is easily scalable.

The rest of this paper is organized as follows. Section II describes the system model, which consists of a network topology and the necessary physical-layer constraints for dense wireless areas. Based on the underlying network model, Section III describes the three-step proposed CCT algorithm. Simulation results as well as the computational complexity of the proposed algorithm appear in Section IV. Section V explains a modified version of the algorithm that permits users with varying rate requirements, and Section VI concludes our work.

II. System Model

Consider a cellular network with U users randomly

distributed within the cell. All users need to communicate with the cellular BS, which is located at the cell's center. Each node $i \in \mathcal{U} = \{1, 2, \dots, U\}$ can act as either a hotspot or a slave; this role is subject to be changed as network or channel conditions change. A hotspot communicates directly with a BS, and a slave must communicate with a BS via a hotspot, as depicted in Fig. 1. Let H and S denote a set of hotspots and a set of slaves, respectively, such that $S \cup H = \mathcal{U}$ and $S \cap H = \emptyset$. A two-layer transmission scheme occurs when BS-hotspot links create "layer-1" connections and hotspot-slave links create "layer-2" connections. Layer-1 connections act as wireless backhaul for the slaves of each hotspot while providing the required data to all necessary hotspots.

The minimum possible amount of bandwidth allocated to a single user is referred to as a "channel," in this paper. A channel can be as small as a subcarrier of an OFDMA network or as small as a resource block in a Long-Term Evolution (LTE) network [12]. A maximum number of N WS channels are available that can be used by the layer-2 connections. All N WS channels might not be available everywhere within the cell, because of the presence of nearby primary users. Hence, some clusters might select only a subset of the N channels for operation. There are a total of M licensed-band channels available that can be used by the layer-1 connections in the proposed scheme. It is assumed that each node can operate on both licensed and WS bands with the same cellular access technology.

Each node i has a minimum rate requirement, denoted by R_i . If a node is a hotspot, then it has to satisfy its own rate requirement as well as those of the slaves that it serves. The average channel gain to noise power ratio across each channel n between any two nodes i and j is denoted by g_{ij}^n . Each cluster can use the set of available WS channels under the condition that it creates insignificant interference on other clusters. Since we are dealing with densely populated areas such as stadiums or conference halls, it is assumed that the users are stationary for a sufficient duration of time for the algorithm to execute and users to communicate with the BS.

This paper's proposed algorithm can be adapted to any locally available under-utilized spectrum; for example, TVWS or the federal-owned WSs of a higher frequency, as discussed in [10]. In both cases, the available channels can be identified through sensing (if devices are capable) or accessing a database of incumbents.

III. Proposed Algorithm: CCT

The proposed CCT algorithm consists of three main steps, as depicted in Fig. 2. Based on the relative distance between the

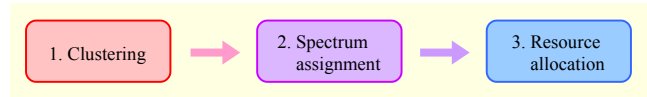


Fig. 2. Proposed three-step CCT algorithm.

nodes or their channel gains, the first step is to cluster the nodes into groups; the BS performs this centrally. The second step is to assign different sets of WS channels to interfering clusters and to try to maximize spectrum reuse. This step is performed in a distributed manner by each cluster based on local channel information. Based on each cluster's assigned set of channels, in step three, the hotspots assign power and specific channels to each slave. Furthermore, in this step, the BS allocates licensed-band resources to the hotspots.

It is assumed that environment conditions are stationary for periods of time after which the algorithm is re-executed. Either the algorithm can be executed at pre-fixed intervals or it can be triggered if there are large changes in the network. However, considering system changes such as number of users (U), user mobility, and other conditions that require modifying cluster size, device handover, and so forth requires a more in-depth study, which is out of the scope of this work.

1. Clustering: Modified k -Means Clustering

The first step clusters the nodes into groups and selects a single node within each cluster to serve as a hotspot. The objective of this step is to group neighboring users together with the constraint that the number of nodes within each cluster is similar. By grouping neighboring nodes together, communication within clusters (hotspot-slave) occurs over shorter distances and, on average, better channels. The second step of CCT assigns equal amounts of spectrum to all clusters; hence, to balance the load among all clusters, an equal number of nodes should be assigned to each cluster.

There are methods for performing such clustering that differ in terms of their objective and complexity [13]. Before selecting a specific method, the problem is formulated mathematically with the required objective and constraints.

We denote with γ the appropriate number of nodes per cluster. The total number of clusters, K , is then equal to $\lceil U/\gamma \rceil$. Clustering nodes into groups consists of identifying the center location of each cluster, $c_k, \forall k \in \{1, \dots, K\}$, and determining the binary indicator variable, $Y_{ik} \in \{0, 1\}, \forall i \in \{1, 2, \dots, U\}, \forall k \in \{1, 2, \dots, K\}$, that indicates if node i belongs to cluster k or not. A method of determining an appropriate value for K and hence γ is explained in Section III. Further, the algorithm assumes the physical location of each node, l_i , is known. Each user device is either equipped with GPS or through localization techniques the BS can determine their location. The clustering

optimization problem can then be formulated as follows:

$$\underset{Y, c}{\text{minimize}} \quad \sum_{i=1}^U \sum_{k=1}^K (Y_{ik} \|l_i - c_k\|_2^2), \quad (1)$$

$$\text{subject to} \quad \sum_{k=1}^K Y_{ik} = 1, \forall i, \quad (2)$$

$$\sum_i Y_{ik} \leq \gamma, \forall k, \quad (3)$$

$$Y_{ik} \in \{0, 1\}, \forall i, k. \quad (4)$$

The objective function (1) minimizes the mean square distance between each node and the center of the cluster. Constraint (2) forces each node to be allocated to a single cluster, and constraint (3) provides an upper bound, γ , for the cluster size. This is an integer programming problem that is NP-hard to solve. However, careful examination of this problem reveals that by eliminating constraint (2), the problem can be solved via the well-known k -means clustering algorithm [14]. k -means clustering is a well-recognized method in data mining for partitioning data into k clusters based on a similarity metric between a data point and a cluster. In the problem at hand, the similarity metric is the distance between each node and the center of the cluster. Such clustering is an NP-hard problem, but with some efficient heuristic methods, it can converge rapidly to local minima. Some variations of the k -means clustering algorithm include adding extra constraints when assigning nodes to clusters [14], [15]. For example, [14] sets a limit on the minimum number of points within each cluster to prevent generating clusters with no data points.

On the other hand, the proposed CCT algorithm introduces an upper bound (γ) on the number of nodes per cluster. This constraint balances the load among clusters and the traffic carried over the hotspot–BS links. Given a set of U nodes with locations $l_i \in [0, L]$, a heuristic method similar to [14] for generating k clusters having an upper bound on the number of nodes per cluster is described in Algorithm 1.

Algorithm 1. Constrained k -means clustering.

Initialize

- $t \leftarrow 1$
- Randomly select cluster centers: $c_k^{(1)} \in [0, L], \forall k \in \{1, 2, \dots, K\}$

repeat

- 1) **Cluster Assignment:** Assign $i \in \{1, \dots, U\}$ to cluster k by solving relaxed vers. of problem (1) given $c_k^{(t)}$.
- 2) **Cluster Update:** Compute $c_k^{(t+1)}$ as the mean of the location of all points assigned to cluster k .

until $c_k^{(t+1)} = c_k^{(t)}, \forall k$

This algorithm, similar to the k -means clustering algorithm

and the constrained k -means clustering algorithm [14], is a heuristic method used to solve biconvex optimization problems.

It consists of (a) fixing cluster centers $c_k, \forall k$ and minimizing over $Y \in R^{U \times K}$, and then (b) fixing Y and minimizing over c . As explained in Algorithm 1, the first step, cluster assignment step, assigns each node i to cluster k by solving optimization problem (1) given cluster centers c_k and relaxing binary variable $Y_{ik}, 0 \leq Y_{ik} \leq 1$. The resulting linear programming problem can be solved via the simplex method with polynomial-time complexity [16]. The second step, cluster update step, updates the center of each cluster by averaging the location of the nodes assigned to it. This step consists of computing the average of γ location values for each cluster.

Similar to the k -means clustering algorithm, this algorithm converges to locally optimal points and is very dependent on cluster center initializations. Hence, Algorithm 1 can be executed multiple times with different initializations and the result with the lowest objective value selected as the final node configuration.

Hotspot selection. The above heuristic, groups the nodes into k clusters. To determine the hotspot within each cluster, one simple approach is to select the node that is closest on average, to both the center of the cluster and the BS. This method of selection, on average, provides better communication links for both layer-1 and layer-2 connections. The algorithm denotes the hotspot of each cluster k by $h_k \in H$ and the set of its slaves by $S_k = \{i | Y_{ik} = 1, i \neq h_k\}$. On average, hotspots consume larger transmission power than slaves. Therefore, the hotspot role can be rotated between the nodes within each cluster.

2. Spectrum Assignment: Graph-Coloring Approach

With the formation of multiple clusters within the cell, spectrum can be reused among them, essentially increasing the amount of usable spectra. In a clustered configuration of nodes, where the transmitters and receivers (hotspots and slaves) are located in close vicinity, lower transmission power is required for communication. The low transmission power allows spectrum reuse among spatially separated clusters.

Many different factors such as the amount of available spectra, user rate-requirements, and node locations affect the spectrum reuse pattern. To determine the reuse, one approach is to create an interference graph of all clusters and assign bands of spectrum based on a graph-coloring approach, similar to the approach studied in interfering femtocell networks [17].

In graph theory, graph coloring is a method of assigning colors to the edges or vertices of a graph while satisfying a set of constraints. For example, in vertex coloring, the vertices of a graph are colored such that no two adjacent vertices share the same color. Two vertices are considered adjacent if they are

connected by an edge. Generally, the objective is to color the graph with the minimum number of colors possible. This number is bounded above by the highest vertex degree of the graph, which is the maximum number of edges attached to a vertex in the graph.

In the spectrum allocation problem at hand, each cluster is represented by a vertex, and two vertices are connected with an edge if they create significant interference on each other. Assume each node is transmitting over its slave-hotspot link at the highest required transmission power to satisfy its rate requirement. If any cluster can receive a signal from another that has power greater than a fraction (α) of the receiver noise power, then the two clusters are connected with an edge.

This paper employs a greedy graph-coloring approach performed in a distributed manner by each cluster, analogous to the distributed algorithm in [18]. Given that the number of colors is greater than the vertex degree of the graph, this algorithm requires $O(\log(K))$ iterations to converge, where K is the number of clusters. We assume the maximum number of channels required by each cluster (β) can be determined by the BS. Then, the algorithm groups every set of β white-space channels together and denotes each such group as a band. If there are a total of N white-space channels available within the cell, then the maximum number of colors available is $B_{\max} = \lfloor N/\beta \rfloor$. The algorithm denotes the set of available bands by $\xi = \{1, 2, \dots, B_{\max}\}$, such that each band or color is identified with a unique number between 1 and B_{\max} .

The greedy graph-coloring works as follows: each hotspot starts by randomly selecting a color from ξ and broadcasting its decision. Each hotspot then listens to the decision of its neighbors (interferers). If it detects a collision between its decision and its neighbor's, then it reselects a color from ξ after eliminating the non-colliding neighbors' colors. This approach reiterates until no two neighbors have the same color. Denote the color of cluster k with b_k and the set of neighboring clusters with η_k . The coloring algorithm performed by each cluster k is then summarized as follows:

Algorithm 2. WS spectrum assignment.

Initialize

- $\xi_k \leftarrow \xi$
- Randomly select b_k from ξ_k

while b_k equals $b_{k'}, \exists k' \in \eta_k$ **do**

- $\xi_k \leftarrow \{b_k\} \cup \xi \setminus \{b_{k'}, \forall k' \in \eta_k\}$
- randomly select b_k from ξ_k

end while

To illustrate the progress of this algorithm, consider a simple example with $K = 5$ clusters (vertices), as shown in Fig. 3. Clusters that interfere with each other and should use different

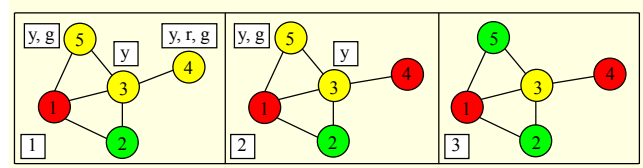


Fig. 3. Distributed graph-coloring example.

sets of spectrum are connected with an edge. In this scenario, for example, node 5 interferes with nodes 1 and 3 if operating on the same spectrum ($\eta_5 = \{1, 3\}$). The highest vertex degree in this graph is “4,” but assume there are only three sets of channels available ($B_{\max} = 3$). These colors are denoted by red (r), green (g), and yellow (y): $\xi = \{r, g, y\}$.

Each cluster initializes its selection pallet ξ_k with all the available colors ($\xi_k \leftarrow \xi, \forall k$). At the first iteration, each cluster (k) selects a color from ξ_k uniformly at random as shown in the left box of Fig. 3. Each cluster (k) then broadcasts its color, b_k , and listens to its neighbors' selections $b_{k'}, \forall k' \in \eta_k$. All clusters that do not detect a collision (1 and 2) fix their selections. A cluster that fixes its selection is labelled “colored.” For the next color selection iteration, the rest of the clusters (3, 4, 5) eliminate their neighbor's colors from ξ (while keeping their own color) and generate the next set of possible colors to select from. The possible colors are shown in white boxes near each vertex in Fig. 3. For example, cluster 5 eliminates {r} from ξ and is left with {y, g}.

During the second iteration, each of the “uncolored” clusters selects a color at random from its set of possible colors. The result of such coloring is shown in the middle box of Fig. 3. Assume clusters 5 and 3 have chosen the same color “y” again, while cluster 4 is successfully colored. The possible choices of 3 and 5 remain the same as before, and during the third iteration, cluster 5 selects “g,” while cluster 3 selects its only option, “y,” as illustrated in the right box of Fig. 3. With this selection, the graph is colored such that no two adjacent clusters are colored with the same color.

If the graphs' vertex degree is less than B_{\max} , then it is guaranteed that Algorithm 2 can successfully assign WS spectra to all clusters. This bound is very loosely based on the sparsity of the interference graph. For example, in the above mentioned five-node example, the graph was colored with 3 colors even though 5 colors are required to guarantee its convergence.

Denote with B^* the minimum number of colors that the greedy algorithm can successfully color the graph. If the number of available bands B_{\max} is greater than the minimum required B^* , then the extra ($B_{\max} - B^*$) bands are wasted when coloring with B_{\max} colors. Instead, if the algorithm is colored with B^* colors, then the remaining bands can be distributed among clusters. Hence at the expense of performing an

iterative graph-coloring algorithm to find B^* , the available WS channels can be better utilized.

Algorithm 2 is said to fail to color all clusters with some \tilde{B} colors if for some number of iterations T_2 (say 3 or 4), the same set of uncolored nodes remain. With this definition, the iterative graph-coloring algorithm with the objective of finding the minimum number of required colors is summarized in Algorithm 3. This algorithm works as follows: Initialize \tilde{B} , the input to Algorithm 2, with some number less than the maximum number of available colors: $(B_{\max} - \delta)$, where $\delta \in \mathbb{Z}^+$ depends on interference graph sparsity. If Algorithm 2 fails with \tilde{B} colors, then add one (or a larger number for faster convergence) to \tilde{B} . Repeat this until Algorithm 2 succeeds or \tilde{B} reaches B_{\max} at which B^* is obtained. If $B^* = B_{\max}$, then more bands than the available WS are required to color the graph. The clusters that are not successfully colored at this point, directly connect to the BS instead of using WSs in a slave-hotspot configuration.

Graph coloring with locally available colors. Each cluster can determine its locally available WSs through either sensing or connecting to a database of registered primary users (or secondary users in the case of radar bands [10]). There are situations where all clusters cannot operate on all the WS channels (ξ), because of the presence of local primary users. In this situation, every cluster k can determine its own set of available bands $\xi_k \subseteq \xi$. Then, Algorithms 2 and 3 can be performed by replacing the general ξ with each cluster's ξ_k .

Algorithm 3. Iterative WS spectrum assignment.

Initialize

- $\tilde{B} \leftarrow B_{\max} - \delta$

while Algorithm (2) fails and $\tilde{B} \leq B_{\max}$ **do**

$\tilde{B} \leftarrow \tilde{B} + 1$

end while

$B^* \leftarrow \tilde{B}$

This initialization is similar to the previous case, where the algorithm is at an iteration when cluster k observes “colored” neighbors with colors in the set $\xi - \xi_k$. In this situation, cluster k 's set of possible bands shrinks to ξ_k .

3. Resource Allocation

The objective of the third step of CCT is minimizing total network transmission power given user rate-requirements. After determining the set of WS channels on which each cluster can operate, CCT allocates specific channels and power resources to all slaves and hotspots. Resource allocation among clusters can be performed in a distributed manner based on local channel information by the hotspots. Licensed-band

resource allocation among hotspots is performed by the BS.

Based on the second step of CCT, each cluster $k \in \{1, \dots, K\}$ is assigned a spectrum band b_k , which consists of β white-space channels. Each cluster's hotspot, h_k , assigns power and channels to the set of its slaves, S_k , similar to OFDMA-based resource allocation, such that each channel is allocated to a single user. Therefore, β should be selected such that it is greater than the number of slaves of each cluster: $\gamma - 1 < \beta$. This step of the algorithm determines for each cluster k , the variables $x_i^n \in \{0, 1\}$ and $p_i^n \in \mathbb{R}_{\geq 0}$, $\forall i \in S_k, \forall n \in b_k$. The indicator variables x_i^n denote if channel $n \in b_k$ is allocated to user $i \in S_k$ or not, and the variables p_i^n indicate the amount of power allocated to user i on channel n . The optimization performed by each hotspot can then be formulated as follows:

$$\text{minimize}_{P, X} \sum_{i \in S_k} \sum_{n \in b_k} \{p_i^n\}, \quad (5)$$

$$\text{s.t.} \sum_{n \in b_k} x_i^n b \log\left(1 + \frac{p_i^n g_{ihk}^n}{x_i^n}\right), \forall i \in S_k, \quad (6)$$

$$\sum_{i \in S_k} x_i^n \leq 1, \forall n \in b_k, \quad (7)$$

$$p_i^n \geq 0, \forall n \in b_k, \forall i \in S_k, \quad (8)$$

$$x_i^n \in \{0, 1\}, \forall n \in b_k, \forall i \in S_k. \quad (9)$$

The objective here is to minimize the transmission power of all slaves S_k over all the available channels b_k , subject to minimum rate requirements identified by constraint (6). Each channel can only be assigned to a single slave, as denoted by constraint (7). Constraint (8) enforces power to be positive, and constraint (9) forces the channel selection variables x_i^n to take on binary values.

A similar optimization problem to that of (5) can be formulated for licensed-band resource allocation among hotspots. In this case, the set of slaves S_k is replaced with the set of all hotspots H , and the set of WS channels b_k is replaced with the set of all licensed bands $\{1, \dots, M\}$. Furthermore, the rate requirement that needs to be satisfied over the hotspot-BS links is equal to the rate requirement of each hotspot node plus the rate requirement of its slaves, hence the right side of inequality constraint (6).

Problem (5) is a mixed integer-programming problem with a combinatorial solution. The integer constraint (9) can be relaxed and solved via dual decomposition as discussed in [19]. However, for dense areas with large number of users, the computational complexity is large. A simpler but non-optimal method is proposed here — one that is similar in approach to the “Proportional Resource Allocation” algorithm of [20]. The

proposed algorithm decomposes the joint optimization of channel assignment and power allocation (5) into two steps. It assigns channels to users and then optimizes power allocation on the assigned channels.

The proposed method (summarized in Algorithm 4 in Appendix A) starts by assigning channels to users sequentially, beginning with the user that has the worst average channel gain to the destination. After assigning channels, power allocation based on optimum water filling for power minimization is applied [21] on the assigned channels. The algorithm then tries to improve the channel assignments by taking channels from the best users (users consuming least amount of power) and giving them to the worst users (consuming large amounts of power), but only if the reassignment reduces the total power. The reassignment continues for a predetermined number of iterations, T_3 . For small number of users and channels (for example, resource allocation within a cluster), the result of the proposed algorithm has been compared to the optimum resource allocation, and the results are identical.

IV. Simulation Results

This section evaluates the performance of the proposed hotspot–slave configuration and the three-step CCT algorithm. Direct communication performance over only licensed bands and over both licensed bands and WS's are used for comparison purposes.

1. Simulation Setup

Consider U users that are randomly distributed in an $L \times L$ square area. All users simultaneously communicate with a BS, which is located at the center of this area and operates on licensed bands for layer-1 communications. Simulations assume the smallest amount of bandwidth that can be allocated to a user (a channel) is $b = 180$ kHz — an amount similar to that allocated to an LTE resource block. Further, simulations assume only 10 MHz of the 3.55 GHz to 3.65 GHz radar band is available and can be used in this region by the nodes as tertiary users. To be compatible with the licensed band allocations, the white-space channels are also 180 kHz wide and are used for layer-2 communications.

To cluster users in step one of CCT, parameter $K = \lceil U / \gamma \rceil$ must be known. The best value of K corresponds to the case that the number of available WS bands, B_{\max} , closely matches B^* , which is the minimum number of colors required to color the clusters. If there is not enough spectrum for all clusters ($B^* > B_{\max}$), then some users will have to directly connect to the BS and utilize the licensed bands (no reuse). If there are more colors than required ($B^* < B_{\max}$), then even though the

excess colors will be distributed among clusters, WS will not be efficiently reused.

To achieve the ideal situation where $B^* \approx B_{\max}$, each cluster is filled with users until there is no longer enough WS for all clusters to utilize. Assuming limited WS resources such that each slave is assigned a single channel, the number of WS channels β per band is equal to $\gamma - 1$ (subtracting a hotspot node). The best choice of γ , and hence K , can then be obtained by approximating the value of B^* (which depends on the sparsity of the interference graph) and setting it equal to $B_{\max} = \lfloor N / (\gamma - 1) \rfloor$.

In the WS spectrum assignment step of CCT, clusters can identify if they interfere with each other by broadcasting pilot signals and identifying interfering neighbors based on the received signals. However, in simulations, to determine if two clusters interfere or not (if there is an edge between two clusters during graph coloring), a distance threshold, d_{th} , is introduced. Two clusters are deemed to be interfering with one another if they are located within a distance d_{th} of each other. Such a parameter depends on various other network parameters, such as the density of nodes, rate requirement, and bandwidth allocated to each user.

To determine d_{th} , the maximum transmission power on white-space channels is calculated based on the rate requirements of users and the average distance between transmitters and receivers. With K clusters in an L^2 area, the average area that a single cluster occupies is L^2/K . Hence, the average distance between a slave and a hotspot within a cluster is approximately half the diameter of the cluster area: $0.5(2L^2/K)^{0.5}$. Assuming each slave has a rate requirement of R , and assuming the worst case that each slave is assigned only one channel, the transmission power can be estimated. Based on the estimated power, d_{th} is obtained as the distance at which the signal power attenuates to αN_0 . In the simulations, α is set to 0.05. A simplified path-loss model [22] with a path-loss exponent of “4” is assumed here, and the noise power per channel bandwidth is set to 10^{-13} W.

It is assumed that environment conditions are stationary for some periods of time after which the algorithm is re-executed. Either the algorithm can be executed at pre-fixed intervals or it can be triggered if there are large changes in the network. However, considering system changes such as number of users (U), user mobility, and other conditions that require modifying, cluster size, device handover, and so forth require a more in-depth study, which is out of the scope of this work.

2. Results and Discussion

The performance of the proposed CCT algorithm is compared to traditional cellular Direct Mode (DM: licensed), where the

Table 1. Selecting optimum K .

γ	2	3	4	5	6
B_{\max}	52	26	17	13	10
B^*	49.0	23.0	14.11	12.22	12.0

BS directly communicates with the nodes over licensed bands. For a fair comparison in terms of available resources, we assume that the BS can also communicate directly with the nodes over WSs as well as licensed bands. However, such an assumption can only be true if the transmission power over the node and BS links on WSs is low enough to ensure no interference effect on the operation of any primary and secondary users. This mode of operation is referred to as “DM: licensed + WS.”

Simulations assume there is a limited amount of available white space in the radar band (the middle 10 MHz, 3.595 GHz to 3.605 GHz), which corresponds to $N = 52$ white-space channels. Simulations further assume that all users have a rate requirement of $R = 540$ kbps [23]. With this set of parameters, Table 1 represents the average minimum number of required WS bands (B^*) when the cluster size γ varies between two and six.

As the relationship $B_{\max} = N/(\gamma - 1)$ dictates, as the cluster size increases, the number of available bands or colors decreases. Furthermore, the minimum required colors to color all clusters decreases, because the clusters are located farther apart and the interference graph is sparser. The optimum K then corresponds to the cluster size γ at which the average B^* is closest to the number of available channels B_{\max} . Table 1 indicates the optimal cluster size ($\gamma = 5$), and by assigning a single channel to each slave, the number of channels per cluster is $\beta = 4$.

Tables (for example, Table 1) that are a function of different system parameters can be stored at the BS. Based on the underlying system parameters, the BS can then use the corresponding look-up table to determine the optimum cluster size K . Keeping the cellular area (L^2) fixed and increasing the number of users, Fig. 4 reveals how increasing the density of users affects the required transmission power in each method. The number of licensed-band channels is set to $M = 500$, which corresponds to 100 MHz of bandwidth (based on LTE advanced carrier aggregation). The number of users is increased from 100 to 500. All results have been obtained by averaging the transmit power values over twenty randomly generated node configurations. Figure 4 indicates that the ratio of required transmission power in CCT relative to “DM: licensed” decreases as the density of users increases.

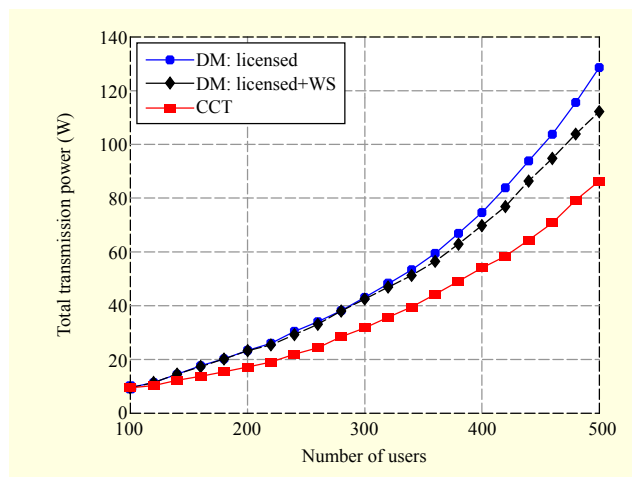


Fig. 4. Network power vs. number of users.

At a user density of 300, the total transmission power required by CCT is about 71% of both DM methods. At a higher density of 500, when there are limited licensed band resources, this ratio reduces to 67% relative to “DM: licensed.” Figure 4 also suggests a 20% increase in the number of users when comparing CCT to both DM approaches. With a total network transmission power of approximately 75W, DM: licensed method can support 400 users, while CCT can support 480 users.

Comparing the two DM methods, when there is an abundance of licensed band spectrum, ($M \gg U$), performance of “DM: licensed + WS” is similar to “DM: licensed.” Licensed band resources operating at lower frequencies relative to WS require lower transmission power and are mainly used in “DM: licensed + WS” operation. The extra WS resources cannot significantly reduce transmission power. When the number of users increases, and spectrum becomes scarce, the extra WS channels are useful in lowering transmission power.

Similar CCT performance results compared to DM methods can be obtained by fixing the density of users and varying the amount of licensed-band channels available. With 500 users, Fig. 5 shows how the required transmission power in each method varies when the number of licensed-band channels increases from 500 (minimum for 500 users) to 700.

As is apparent from Fig. 5, the required transmission power decreases as the number of licensed-band channels increases.

When there is limited licensed spectrum available ($M = 500$), the total transmission power required by CCT is only 66% of “DM: licensed” and 77% of “DM: licensed + WS” method. As M increases, the performance of “DM: licensed + WS” approaches “DM: licensed,” because the WS channels at much higher frequencies than licensed channels require higher transmission power and hence are not as effective. This figure

Table 2. Ratio of CCT to DM power given path-loss exponent.

Pathloss exponent	3	3.5	4	4.5	5
CCT/DM power (%)	98	87	68	60	56

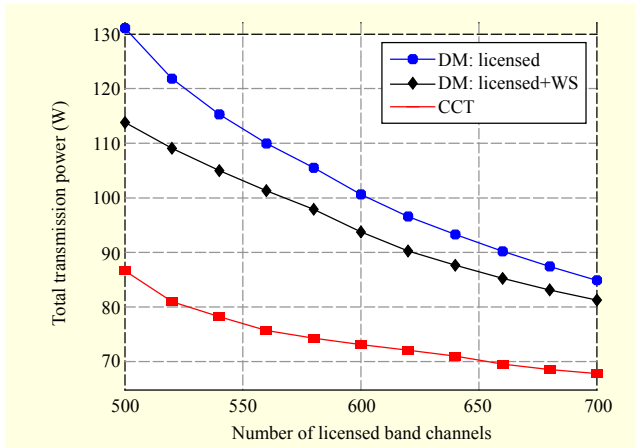


Fig. 5. Network power vs. number of licensed-band channels.

shows that the proposed CCT algorithm can reduce the required network resources at the expense of the hotspot–slave configuration overhead.

Assuming a fixed density of users ($U = 500$), and taking $M = 500$, Fig. 6 presents the total network transmission power when the per-user rate requirement increases from 0.18 Mbps to 1.08 Mbps. As the user rate-requirements increase, the distance threshold d_{th} increases, and hence more WS channels are required. Simulation results show that the average number of WS channels required by CCT for each rate requirement varies from $N = 26.8$ to 116.0. The same number of channels are used for “DM: licensed + WS” for comparison. The ratio of transmission power for CCT relative to “DM: licensed + WS” for all rate requirements is fixed around 0.75. However, the ratio of transmission power used by CCT decreases relative to “DM: licensed” from 0.74 to 0.51.

3. Performance Evaluation

Two important factors that affect the performance of CCT relative to DM are the channel path-loss exponent and the WS operating frequency. With lower channel path-loss exponent, the transmitted signal is less attenuated with distance; hence, it can also propagate better with distance. In the proposed multi-hop configuration, the performance gain of CCT is higher compared to direct communication as the signal attenuation with distance is higher. The reason for this is that when the signal is attenuated enough (high path loss), there are two consequences: (a) a signal originating within a cluster creates

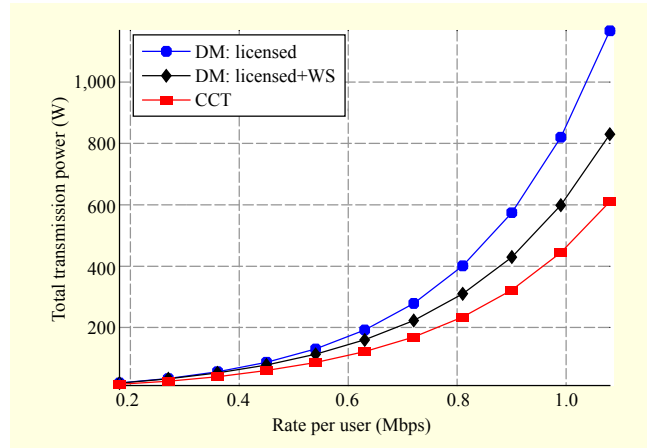


Fig. 6. Network power vs. per-user rate requirement (500 users).

less interference on neighboring clusters and spectrum reuse can increase and (b) the effect of path loss reduction is more pronounced when using two hops instead of one in direct communication.

Table 2 indicates how the performance of CCT varies by increasing the channel path-loss exponent from three to five in increments of 0.5. At a path-loss exponent of three, CCT utilizes almost the same amount of transmission power (98%) as direct communication. At a path-loss exponent of five (found in urban areas), CCT utilizes only 56% of power used in direct mode. Employing WSs operating at lower frequencies, such as TVWS (50 MHz to 698 MHz) [8], the CCT gain degrades relative to the second direct-mode method, “DM: licensed + WS.” Simulations have been performed for the case where clusters utilize TVWS channels 22 and 24, operating on 518 MHz to 524 MHz and 530 MHz to 536 MHz bands, respectively. The WS channels are divided such that there are also $N = 52$ channels.

Other than the WS operating frequencies, with similar setup as Fig. 5 and with 500 users and 500 licensed bands, CCT utilizes 66% of “DM: licensed” transmission power. This is in comparison to the 77% in Fig. 5. The reason for this lower transmission power ratio is that less power is required for gigahertz-federal-owned bands. CCT utilizes 94% of the power required for “DM: licensed + WS.” This implies that when TVWS is available to be used along the longer links from the BS to nodes, it suffices to perform direct communication over licensed and WS bands. This is only when TVWS can be utilized over longer ranges without creating interference on primary users.

4. Computational Complexity

This section investigates the complexity of implementing the three-step CCT algorithm. The first step, clustering, consists of

$O(\log(U))$ iterations of (a) solving an LP with complexity $O(U^m)$ (m is a constant) and (b) K -mean calculations of γ numbers. An iteration consists of T_2 iterations of Algorithm 2, which has complexity $O(\log(K))$. The number of iterations (T_2) required to find B^* depends on how close the initial value of \tilde{B} is to B^* . The third step consists of K resource allocations for a maximum of γ users and a single resource allocation for K hotspots. The resource allocation algorithm consists of sorting and basic arithmetic operations, such as averaging and comparison, which have polynomial time complexity.

V. Unequal Rate Requirements and Virtual Nodes

The CCT algorithm's clustering step considers equal rate requirements for all users. With a simple modification, the algorithm can also be applied to a network of users with different rate requirements. To solve this issue while keeping the already proposed algorithms tractable, the concept of virtual nodes is proposed. Some number of virtual nodes (with equal rate requirements) is introduced for every single node in the system such that the sum of the rate requirements of all the virtual nodes is equal to the rate requirement of an actual node. For this purpose, a minimum user rate-requirement, r_{\min} , is identified. Then, every user i 's rate requirement, R_i , can be rounded to an integer multiple \hat{R}_i of r_{\min} , (that is, $R_i = \hat{R}_i r_{\min}$). Hence, instead of user i , \hat{R}_i virtual nodes are substituted in the network.

In the modified k -means clustering algorithm, the number of nodes in each cluster k (that is, Y_{ik}) is limited by γ as in equation (7). To add in the virtual nodes, one can multiply each link variable Y_{ik} with node i 's equivalent number of virtual nodes. This is done by replacing equation (7) with $\sum_i Y_{ik} \hat{R}_i \leq \gamma$,

$\forall k$. Other than substituting each node with multiple virtual nodes, this modification does not affect the proposed CCT algorithm.

VI. Summary

The future is ubiquitous wireless access anywhere and at any time — be it a conference hall, a sporting event, or any other type of dense gathering. This paper proposes coordinated tethering over WSs in densely populated areas. The proposed algorithm, CCT, can improve spectrum utilization with low cost and complexity by configuring some nodes to serve as hotspots and using the otherwise unused WS bands for data offload. By employing CCT, up to 30%, on average, increase in device battery life can be obtained as compared to conventional direct-mode cellular communication. Given fixed

network resources, CCT can increase the number of active users by 20%. CCT can provide simultaneous service to more users than available licensed channels. All these benefits are gained at the expense of clustering, graph-coloring overhead, and hotspot nodes consuming more power than their regular direct connection.

Appendix: Resource Allocation among Nodes

The algorithm used to allocate resources among clusters by each hotspot as well as among hotspots by the BS is summarized in this section. Assume there are M channels that are to be assigned to U users. One set of inputs to this algorithm are the channel gains of all users over all M channels denoted by the matrix $\mathbf{G} \in \mathbb{R}^{U \times M}$. If allocating resources among a cluster k , \mathbf{G} will contain the channel gains between all the slaves S_k and the hotspot H_k , and if allocating resources among hotspots, \mathbf{G} will contain channel gains between hotspots and the BS. The second set of inputs is the rate requirement of all users denoted by the vector $\mathbf{r} \in \mathbb{R}^U$. The output of the algorithm is the power allocation matrix $\mathbf{P} \in \mathbb{R}^{U \times M}$.

Channel assignment parameters x_n^i are related to the power parameters p_n^i such that $p_n^i \geq 0$ if $x_n^i = 1$ and $p_n^i = 0$ if $x_n^i = 0$. The parameters x_i , g_i , and p_i without the superscript denote a vector of values for user i over all channels. For an even distribution of resources, with U users and M channels, there are $\lfloor M/U \rfloor$ channels per user with $(U - U \lfloor M/U \rfloor)$ extra channels. Algorithm 4 starts by sorting users according to their average gains over all channels. Then, starting from the user with the worst average channel gain, up to $(U - U \lfloor M/U \rfloor)$ users, it assigns $(\lfloor M/U \rfloor + 1)$ channels to users sequentially. The remaining users then obtain $\lfloor M/U \rfloor$ channels each. Such distribution allows a relatively fair resource distribution such that the per-user required transmission powers are comparable.

After assigning channels, power allocation based on optimum water filling for power minimization is applied [21]. This optimization is represented by the function $\text{WF}(\cdot)$, which takes as input user i 's rate requirement and the gains of the channels assigned to user i , and outputs per-channel power p_i .

The algorithm then tries to improve channel assignments by taking channels from the best users (consuming least power) and giving them to the worst users (consuming most power), but only if the reassignment reduces the total power. Set V determines the set of nodes that can relinquish a channel to other users. Since every node needs to have at least one channel, set V is initialized to nodes that have more than one channel. Denote the user that requires the least power by u_{\min} and the user that requires the most by u_{\max} .

Algorithm 4. Resource allocation.

Initialize

Iteration number: $t \leftarrow 1$

Channel assignment: $x_i^{(1)}$

- Sort user based on average channel gains
- Sequentially assign channels starting from worst user
Power allocation: $p_i^{(1)} = \text{WF}(x_i^{(1)}, g_i, r_i), \forall i \in \{1, \dots, U\}$
Valid set: $\mathcal{V}^{(1)} \leftarrow \{i \mid \sum_n x_i^n > 1\}$

while $\mathcal{V} \neq \emptyset$ and $t < T_3$ **do**

$u_{\max} \leftarrow \arg \max_u \{\sum_n p_u^n\}$

$u_{\min} \leftarrow \arg \min_{u \in \mathcal{V}} \{\sum_n p_u^n\}$

$\tilde{x}_{\min} \leftarrow x_{u_{\min}}, \tilde{x}_{\max} \leftarrow x_{u_{\max}}$

$\tilde{x}_{\min}^{\tilde{n}} \leftarrow 0, \tilde{x}_{\max}^{\tilde{n}} \leftarrow 1$

$\tilde{p}_{\min} \leftarrow \text{WF}(\tilde{x}_{\min}, g_{u_{\min}}, r_{u_{\min}})$

$\tilde{p}_{\max} \leftarrow \text{WF}(\tilde{x}_{\max}, g_{u_{\max}}, r_{u_{\max}})$

if $\tilde{p}_{\max} + \tilde{p}_{\min} > p_{u_{\min}}^{(t)} + p_{u_{\max}}^{(t)}$ **then**

$\mathcal{V}^{(t+1)} \leftarrow \mathcal{V}^{(t)} \setminus \{u_{\min}\}$

else

$x_{u_{\min}}^{(t+1)} \leftarrow \tilde{x}_{\min}, x_{u_{\max}}^{(t+1)} \leftarrow \tilde{x}_{\max}$

$p_{u_{\min}}^{(t+1)} \leftarrow \tilde{p}_{\min}, p_{u_{\max}}^{(t+1)} \leftarrow \tilde{p}_{\max}$

end if

if $\sum_n x_{u_{\min}}^n = 1$ **then**

$\mathcal{V}^{(t+1)} \leftarrow \mathcal{V}^{(t)} \setminus \{u_{\min}\}$

end if

end while

$t \leftarrow t + 1$

The channel (\tilde{n}) that has the highest channel gain for the receiving node is selected to be removed from u_{\min} and given to u_{\max} . Based on this reassignment, the updated channel assignments and updated required transmission powers for nodes u_{\min} and u_{\max} are temporarily stored in x and p , respectively. If the sum of the resulting transmission powers is less than the sum before reassignment, then the reassignment becomes permanent; otherwise, the reassignment does not occur and u_{\min} is removed from the pool of possible donors \mathcal{V} . The reassignment of channels continues for a predetermined number of iterations T_3 or until \mathcal{V} is empty.

Acknowledgement

This paper was submitted in part to IEEE global communications conference, Atlanta, GA, December 2013.

References

[1] C. Lee and H. Kang, "Cell Planning With Capacity Expansion in

Mobile Communications: A Tabu Search Approach," *IEEE Trans Veh. Technol.*, vol. 49, no. 5, Sept. 2000, pp. 1678–1691.

[2] A. Khandekar et al., "LTE-Advanced: Heterogeneous Networks," *European Wireless Conf.*, LuCCA, Italy, 2010, pp. 978–982.

[3] *Cisco Connected Stadium*, Cisco Systems, 2010. Accessed Feb. 2013. http://www.cisco.com/web/strategy/sports/connected_stadium.html

[4] Ruckus Wireless, Inc., *Deutch Telekom Installs State-of-the-Art 802.11n Wi-fi System in one of Germany's Largest Stadiums*, Ruckus Wireless, 2010. Accessed May 2013. <http://www.ruckuswireless.com/press/releases>

[5] Craig Lioyd *Qualcomm Bringing Wifi Improvements to MLB Stadiums*, Qualcomm, 2013. Accessed July 2013. <http://www.slashgear.com/qualcomm-bringing-wifiimprovements-to-mlb-stadiums-04276474/>

[6] H. Tabrizi, G. Farhadi, and J.M. Cioffi, "Casra: An Algorithm for Cognitive Tethering in Dense Wireless Areas," *Proc. IEEE Global Commun. Conf.*, Atlanta, DA, USA, 2013, pp. 3855–3860.

[7] T. Yucek and H. Arslan, "A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications," *IEEE Commun. Surveys Tutorials*, vol. 11, no. 1, Mar. 2009, pp. 116–130.

[8] FCC-10–174, "Before the Federal Communications Commission; Washington, D.C. 20554," Sept. 2010.

[9] R. Saruthirathanaworakun, J. Peha, and L. Correia, "Opportunistic Sharing between Rotating Radar and Cellular," *IEEE J. Sel. Areas Commun.* vol. 30, no. 10, Nov. 2012, pp. 1900–1910.

[10] J.P. Holdren et al., *Report to the President Realizing the Full Potential of Government Held Spectrum to Spur Economic Growth*, PCAST, 2012. Accessed Jan. 2013. <http://www.whitehouse.gov/sites/default/files/microsites/ostp/pcast/spectrum-report-final-july-20-2012.pdf>

[11] T. Todd and D. Zhao, "Cellular CDMA Capacity in Hotspots with Limited Ad Hoc Relaying," *Proc. IEEE Pers. Indoor Mobile Radio Commun.*, Beijing, China, 2003, pp. 2828–2832.

[12] 3GPP-TS-36–213, *LTE: Evolved Universal Terrestrial Radio Access; Physical Layer Procedures*, Cedex, France, 2011.

[13] J. Yu and P. Chong, "A Survey of Clustering Schemes for Mobile Ad Hoc Networks," *IEEE Commun. Surveys Tutorials*, vol. 7, no. 1, May 2005, pp. 32–48.

[14] P. Bradley, K. Bennett, and A. Demiriz, "Constrained K -Means Clustering," Technical Report MSR-TR-2000–65, Microsoft Research, Redmond, WA, USA.

[15] K. Wagstaff et al., "Constrained K -Means Clustering Background Knowledge," *ICML. Morgan Kaufmann*, 2001, pp. 577–584.

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

[17] L. Tan et al., "Graph Coloring Based Spectrum Allocation for Femtocell Downlink Interference Mitigation," *IEEE Wireless Commun. Netw. Conf.*, Cancun, Mexico, 2011, pp. 1248–1252.

- [18] D.A. Grable and A. Panconesi, "Fast Distributed Algorithms for Brooksvizing Colourings," *Proc. Annu. ACM-SIAM Symp. Discrete Algorithms*, Mar. 1998, pp. 473–480.
- [19] M. Mohseni, R. Zhang, and J.M. Coiffi, "Optimized Transmission for Fading Multiple-Access and Broadcast Channels with Multiple Antennas," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, Aug. 2006, pp. 1627–1639.
- [20] I.C. Wong et al., "A Low Complexity Algorithm for Proportional Resource Allocation in OFDMA Systems," *IEEE Workshop Signal Process. Syst.*, Austin, TA, USA, Oct. 2004, pp. 1–6.
- [21] J.M. Coiffi, *Digital Communication*, Stanford University, CA, USA, 2002. Accessed Sept. 2013. <http://www.stanford.edu/group/coiffi/book/chap4.pdf>
- [22] A. Goldsmith, "*Wireless Communication*," New York, NY, USA: Cambridge University Press, 2005.
- [23] Ruckus Wireless, *Deploying Very High Density Wi-fi: Design and Configuration Guide for Stadiums*, Ruckus Wireless, Inc, 2012. Accessed Oct. 17, 2013. <http://www.ruckuswireless.com/carriers/high-density>



Haleh Tabrizi received her BS degree in electrical engineering from the University of California at Los Angeles, USA, in 2007 and her MS and PhD degrees in electrical engineering from Stanford University, Los Angeles, CA, USA, in 2009 and 2013, respectively. She was the recipient of an NSF graduate research fellowship

between 2007 and 2010, and received research grants from Agilent Technologies Santa Clara, CA, USA, Fujitsu Labs of America, Sunnyvale, CA, USA, and the deanship of King Abdulaziz University, Jeddah, Saudi Arabia, between 2010 and 2013. Her research interests include signal processing for wireless communications; non-volatile memory characterization and management; convex optimization; and machine learning.



Golnaz Farhadi received her PhD degree in electrical and computer engineering from the University of Alberta, Edmonton, Canada and subsequently tenured a National Science and Engineering Research Council postdoctoral fellowship with the Department of Electrical Engineering, Stanford University, Los Angeles,

CA, USA. Her research interests include machine learning, optimization, and resource management.



John Matthew Coiffi received his BS degree in electrical engineering from the University of Illinois at Urbana–Champaign, IL, USA, in 1978 and his PhD degree in electrical engineering from Stanford University, Los Angeles, CA, USA, in 1984. He was with Bell Laboratories from 1978 to 1984 and IBM

Research from 1984 to 1986. Since 1986, he has been a professor of electrical engineering at Stanford University, now emeritus. He founded Amati Com. Corp., San Jose, CA, USA in 1991 (purchased by TI in 1997) and was officer/director from 1991 to 1997. He is currently also on the board of directors of ASSIA, Redwood, CA, USA (Chairman and CEO), Alto Beam, Beijing, China, and the Marconi Foundation. Mountain View, CA, USA. His research interests include high-performance digital transmission. He is also an adjunct professor of computing/information technology at King Abdulaziz University, Jeddah, Saudi Arabia. He is the recipient of the IEEE's Alexander Graham Bell and Millenium Medals (2010 and 2000); Member Internet Hall of Fame (2014); Economist Magazine 2010 Innovations Award; International Marconi Fellow (2006); Member, US National and UK Royal Academies of Engineering (2001, 2009); IEEE Kobayashi and Kirchmayer Awards (2001 and 2014); IEEE Fellow (1996); IEE JJ Tomson Medal (2000); 1991 and 2007 IEEE Comm. Mag. best paper; and numerous conference best-paper awards. He has published over 600 papers and holds over 100 patents, of which many are heavily licensed including key necessary patents for the international standards in ADSL, VDSL, DSM, and WiMAX.



Ghadah Aldabbagh received her BS degree in electrical engineering from the University of Illinois at Urbana–Champaign, USA, in 1991; her MS degree in data communication networks and distributed systems from University College London, UK, in 1999; and her PhD degree in electronic and electrical engineering from

University College London, UK, in 2010. She holds an assistant professor position with the Faculty of Computing and Information Technology, Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests include communication systems and networks; network management; large scale multiservice IP networks (fixed and mobile) from the viewpoint of network resourcing; inter-domain QoS; network measurement; complexity issues; security aspects; policy-based management; multimedia networking; networking algorithms and performance evaluation; communication and information theory; protocols and algorithms; networking theory; QoS management; and wireless communications. During her PhD study at UCL, she worked on several European Union projects in grid computing.