This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE SECON 2005 proceedings.

Distributed Spectrum Allocation via Local Bargaining

Lili Cao
Shanghai Jiaotong University, Shanghai, P.R. China
Email: cao_li_li@yahoo.com

Haitao Zheng
Microsoft Research Asia, Beijing, P.R. China
Email: htzheng@ieee.org

***Abstract*— In this paper, we present an adaptive and distributed approach to spectrum allocation in mobile ad-hoc networks. We propose a local bargaining approach where users affected by the mobility event self-organize into bargaining groups and adapt their spectrum assignment to approximate a new optimal assignment. The number of computations required to adapt to topology changes can be significantly reduced compared to that of the conventional topology-based optimizations that ignore the prior assignment. In particular, we propose a *Fairness Bargaining with Feed Poverty* to improve fairness in spectrum assignment and derive a theoretical lower bound on the minimum assignment each user can get from bargaining for certain network configurations. Such bound can be utilized to guide the bargaining process. We also show that the difference between the proposed bargaining approach and the true optimal approach is upper-bounded. Experimental results demonstrate that the proposed bargaining approach provides similar performance as the topology-based optimization but with more than 50% of reduction in complexity.**

I. Introduction

Wireless devices are becoming ubiquitous, placing additional stress on the fixed radio spectrum available to all access technologies. To eliminate interference between different wireless technologies, current policies allocate a fixed spectrum slice to each wireless technology. This static assignment prevents devices from efficiently utilizing allocated spectrum, resulting in spectrum holes (no devices in area) and very poor utilization [16]. These results further motivate the *Open Spectrum* [3] approach to spectrum access. Open Spectrum allows unlicensed (secondary) users to coexist with legacy (primary) spectrum holders, thereby "creating" new capacity and commercial value from existing spectrum ranges. Secondary users opportunistically utilize unused licensed spectrum on a non-interfering or leasing basis based on agreements and constraints imposed by primary users.

Open spectrum focuses on controlling the behavior of secondary users while keeping the system transparent to primaries. While maximizing spectrum utilization is the primary goal of dynamic spectrum systems, a good allocation scheme is also needed to provide fairness across users. We hereby use user to represent secondary user. A user seizing spectrum without coordinating with others can cause harmful interference with its surrounding neighbors, and thus reducing available spectrum. Given a fixed topology, existing approaches can efficiently allocate spectrum to users by reducing the problem to a variant of the graph coloring problem [23], [18]. A conflict free spectrum assignment is obtained for the given topology. In general, a topology-optimized allocation algorithm begins with no prior information, and assigns each user an optimal assignment. In a mobile network, however, users are constantly moving and the network topology changes. Using this global optimization approach, the network needs to completely recompute spectrum assignments for all users after each change, resulting in high computational and communication overhead. This costly operation needs to be repeated frequently to maintain spectrum utilization and fairness.

In this work, we consider a distributed approach to spectrum allocation that starts from the previous spectrum assignment, and performs a limited number of computations to adapt to recent topology changes. We propose a local bargaining approach where users affected by the mobility event self-organize into bargaining groups and adapt their spectrum assignment to approximate a new optimal conflict free assignment. The key contributions of this paper are three-fold:

Local Bargaining Strategy. We propose a local bargaining framework, and two bargaining strategies: *one-to-one fairness bargaining* and *feed poverty bargaining*, to improve fairness based system utility.

Bound of Local Bargaining Performance. We derive a theoretical lower bound on the number of channels each user can get from bargaining, referred to as *Poverty Line*. The bound reflects the level of fairness enforced by the proposed bargaining strategies, and can serve as a guidance for bargaining. We also derive a upper bound on the performance difference between the local bargaining and the global optimal solution, referred to as *price of anarchy*.

Simulation of Efficiency and Complexity. We conduct extensive simulations to quantify the performance of local bargaining. Results indicate the proposed bargaining performs similarly to the graph-coloring solution[23], [18] but with significantly reduced algorithm complexity. We also validate the correctness of the *poverty line* and the effectiveness of *poverty guided bargaining*.

The rest of the paper is organized as follows. We begin in Section II by defining the spectrum allocation problem and the system utility functions. Next, we propose a local bargaining framework in Section III and develop specific strategies to improve system fairness in Section IV. Next in Section V, we conduct experiments to evaluate the performance of bargaining strategy and validate the theoretical lower bound. We then derive analytical results in terms of price of anarchy in

0-7803-9011-3/05/$20.00 ©2005 IEEE.

Section VI. Finally, we summarize related work in Section VII, discuss implications and future directions and conclude in Section VIII.

II. Background on Spectrum Allocation

The two key goals of a spectrum allocation algorithm are spectrum utilization and fairness. Combinations of these two goals form specific utility functions that can be customized for different types of network applications. As background, we describe in this section our previous work on efficient and globally optimized spectrum allocation. We start with the theoretical model used to represent the general allocation problem, and two utility functions that maximize spectrum utilization and fairness. We describe how to reduce the optimal allocation problem to a variant of a graph multi-coloring problem and describe the previous solution that optimize the spectrum allocation for a given topology.

A. Problem Model and Utility Functions

We consider the case where the collection of available spectrum ranges forms a spectrum pool, divided into non-overlapping orthogonal channels. We assume a network of N users indexed from 0 to $N-1$ competing for M spectrum channels indexed 0 to $M-1$. Each user can be a transmission link or a broadcast access point. Users select communication channels and adjust transmit power accordingly to avoid interfering with primaries. The channel availability and throughput for each user can be calculated based on the location and channel usage of nearby primaries. The spectrum access problem becomes a channel allocation problem, *i.e.* to obtain a conflict free channel assignment for each user that maximizes system utility. The key components of our model are:

- Channel availability $L(n)$.
$\Gamma = \{l_{m,n} | l_{m,n} \in \{0,1\}\}_{M\times N}$ is a M by N binary matrix representing the channel availability: $l_{m,n} = 1$ if and only if channel m is available at user n. In general, $l_{m,n} = 0$ when channel m is occupied by a primary user who conflicts with user n, so that the transmissions of n on this channel will interfere with the primary's activity if they use channel m concurrently. Let $L(n) = \{0 \leq m \leq M-1 | l_{m,n} = 1\}$ be the set of channels available at n.

- Interference constraint C.
Let $C = \{c_{n,k} | c_{n,k} \in \{0,1\}\}_{N\times N}$, a N by N matrix, represents the interference constraints among users. If $c_{n,k} = 1$, users n and k would interfere with each other if they use the same channel. The interference constraint depends on the signal strength of transmissions and the distance between users. A simple model of interference constraint is the binary geometry metric, *i.e.* two transmissions conflict if they are within π distance from each other. This provides an approximation to the effects of interference in real wireless systems.

It should be noted that the interference constraint could also depend on the frequency location of the channel (*i.e.* m), since power and transmission regulations vary significantly across frequencies. The work in [23], [18] considers the channel-dependency and uses a M by N by N interference matrix C. In this paper, for simplicity, we consider a channel-independent interference constraint, assuming channels have similar power and transmission regulations. It is straightforward to extend the proposed approaches to account for channel-dependent or other interference conditions [4].

- User dependent channel throughput B.
Let $B = \{b_{m,n} > 0\}_{M\times N}$ describe the reward that a user gets by successfully acquiring a spectrum band, i.e. $b_{m,n}$ represents the maximum bandwidth/throughput that user n can acquire through using spectrum band m (assuming no interference from other neighbors). Let $b_{m,n} = 0$ if $l_{m,n} = 0$. So that B represents the bandwidth weighted user available spectrum.

- Conflict free assignment A.
$A = \{a_{m,n} | a_{m,n} \in \{0,1\}\}_{M\times N}$ where $a_{m,n} = 1$ denotes that spectrum band m is assigned to user n, otherwise 0. A satisfies all the constraints defined by C, that is,

$$a_{m,n} + a_{m,k} \leq 1,\ if\ c_{n,k} = 1, \forall\, n, k < N, m < M.$$

Let $\Lambda_{N,M}$ denote the set of conflict free spectrum assignments for a given set of N users and M spectrum bands.

- User throughput of a conflict free assignment.
Let $TP_A(n)$ represent the throughput that user n gets under assignment A, *i.e.* $TP_A(n) = \sum_{m=0}^{M-1} a_{m,n} \cdot b_{m,n}$.

Given this model, the goal of spectrum allocation is to maximize network utilization, defined by U. We can define the spectrum assignment problem by the following optimization function:

$$A^* = \max_{A\in\Lambda_{N,M}} arg\max U(A),$$

We can obtain utility functions for specific application types using sophisticated subjective surveys. An alternative is to design utility functions based on traffic patterns and fairness inside the network. In this paper, we consider and address fairness based system utility. Consistent with prior work[17], [12], [21], we address fairness for single-hop flows since they are the simplest format in wireless transmissions. We postpone the discussion of routing related utility functions to a future paper. Similar to [17], we define fairness in terms of maximizing total logarithmic user throughput, refereed to as proportional fairness. The utility can be expressed as

$$U(A) = \sum_{n=0}^{N-1} \log TP_A(n) = \sum_{n=0}^{N-1} \log \sum_{m=0}^{M-1} a_{m,n} \cdot b_{m,n}. \quad (1)$$

As a reference, another utility function is the total spectrum utilization in terms of total user throughput, $U(A) = \sum_{n=0}^{N-1} TP_A(n)$. Maximizing utilization does not consider fairness, and the resulting channel assignment is in general unbalanced.

B. Color-Sensitive Graph Coloring

In [23], [18], it is shown that by mapping each channel into a color, the channel assignment problem can be reduced to a graph multi-coloring (GMC) problem.

Definition 1: Given the channel assignment problem in above, the system can be represented by a *Conflict Graph* $G = (V, E, B)$ where V is a set of vertices denoting the users that share the spectrum, B represents the bandwidth weighted available spectrum, mapping to the color list at each vertex, and E is a set of undirected edges between vertices representing interference constraint between two vertices defined by C. For any two distinct vertices u,$v \in V$, an edge between u and v, is in E if and only if $c_{u,v} = 1$.

Fig. 1 illustrates an example of GMC graph. There are 5 colors available. The numbers outside the brackets attached to each node denote the colors assigned to that node, while the numbers inside the brackets denote the available color list of each node.

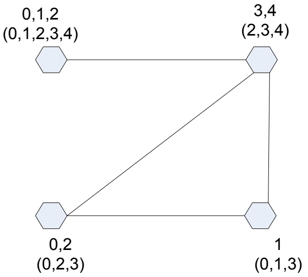


Fig. 1. An example of GMC graph

A GMS problem is to color each vertex using a number of colors from its color list, and find the color assignment that maximizes system utility. The coloring is constrained by that if an edge exists between any two distinct vertices, they can't be colored with the same color. Most importantly, the objective of coloring is to maximize system utility. This is different from traditional graph color solutions that assign one color per vertex. Notice that the solution to this graph coloring problem is to maximize system utility for a given graph, *i.e.* a given topology and channel availability. This characterizes the optimal solution for a static environment.

The optimal coloring problem is known to be NP-hard [7]. Efficient algorithms to optimize spectrum allocation for a given network topology exist. In [23], the authors presented a set of sequential heuristic based approaches that produce good coloring solutions. The algorithm starts from empty color assignment and iteratively assign colors to vertices to approximate the optimal assignment. In each stage, the algorithm labels all the vertices with a non-empty color list according to some policy-defined labeling. The algorithm picks the vertex with the highest valued label and assigns the color associated with the label to the vertex. The algorithm then deletes the color from the vertex's color list, and from the color lists of the constrained neighbors. The color list and the interference constraint of a vertex keep on changing as other vertices are processed, and the labels of the colored vertex and its neighbor vertices are modified according to the new graph. The algorithm can be implemented using a centralized controller who observes global topology and makes decisions, or through a distributed algorithm where each vertex performs a distributed voting process. Results in [18], [23] show that the heuristic based algorithms perform similarly to the global optimum (derived off-line for simple topologies), and the centralized and distributed algorithms perform similarly.

III. Local Bargaining Framework

The approach described in Section II globally optimizes spectrum allocation for a given topology. In a mobile network model, node movements lead to constant changes in network topology. Using the existing approach, we can reapply the spectrum allocation algorithm after each change in the conflict graph. This approach assumes no prior allocation information, and incurs high computation and communication overheads. To reduce these overheads, we propose the use of an adaptive and robust distributed algorithm that takes prior allocation into account in new spectrum assignments.

An efficient dynamic allocation algorithm can run every time user movement causes a change in the corresponding network conflict graph. Therefore, an adaptive algorithm needs to only compensate for small changes affecting a local network region. The algorithm starts from a non-optimal spectrum allocation, which can be constructed from the allocation prior to the topology change. Consider a conflict graph with N nodes (indexed from 0 to $N-1$) and M channels (indexed from 0 to $M-1$), where the optimized assignment is $A_{M\times N}$. When a new node (indexed as N) joins the network, the assignment after introduction of the node becomes $A'_{M\times(N+1)}$ where

$$A'_{m,n} = \begin{cases} A_{m,n} & : \ 0 \le n \le N-1,\ 0 \le m < M \\ 0 & : \ n = N,\ 0 \le m < M \end{cases}$$

Or if a primary user i enters the network and wants to use channel m_0, the nodes within impact of primary i (denoted by $Nbr(i)$) need to stop using channel m_0 within a given time. Hence, the assignment becomes $A'_{M\times N}$ such that

$$A'_{m,n} = \begin{cases} 0 & : \ m = m_0 \ and \ n \in Nbr(i) \\ A_{m,n} & : \ otherwise \end{cases}$$

Assuming the spectrum allocation was near optimal before the topology change, local bargaining between affected vertices can quickly optimize allocations for utilization and fairness. During local bargaining, sets of neighboring vertices, each of which form a connected component of the conflict graph, self-organize into bargaining groups. Each group modifies spectrum assignment within the group to improve system utility while ensuring that the change in spectrum assignment does not require any change at other nodes outside the group (due to interference constraints). Note that a node can represent a transmission link or an access point. Bargaining related to a transmission link is carried out by the transmitter or receiver while bargaining related to an access point is carried out by the access point.

A. Bargaining constraints

To perform bargaining, we must first determine the size and membership of local bargain groups. Large groups increase the complexity of bargaining due to high synchronization and communication costs. In addition, interactions might occur between bargain groups if they share neighboring users. To facilitate bargaining, we propose two constraints to regulate the procedure and simplify the process.

Constraint 1: Limited Neighbor Bargaining.
While pair-wise bargaining is already a hard convex optimization problem, bargaining within a large group implies even higher computation and communication overheads. Coordinating around a central leader per group can greatly simplify spectrum assignment. In this paper, we propose a simple group formation where a node who wants to improve its spectrum assignment broadcasts a bargaining request to its k-hop neighbors, where k is the ratio of interference range and transmission range. These neighbors are connected to the node in the corresponding conflict graph. Those neighbors whose are willing to participate reply to the sender and form a bargaining group. Note that it is possible that two connected neighbors in a conflict graph might not be able to communicate directly with each other, *i.e.* when $k > 1$. The bargaining information can be relayed by the nodes in between. These relay nodes do not necessary participate in the bargaining group. In this following, we will use node to present any vertex in a conflict graph.

For each bargaining group, the requester becomes the group coordinator and performs the bargaining computation. The bargaining strategies can be divided into the following formats:

1.a) One-to-one bargaining- The node n_1 who initiates the bargaining can choose to bargain with only one neighboring node n_2 at a time. They exchange some channels to improve system utility while complying with the conflict constraints from the other neighbors. This is the simplest bargaining process and the requester only needs approval from one of his neighbors to perform the bargaining. When multiple neighbors *e.g.* n_2 and n_3 acknowledge the bargaining request, n_1 can sequentially compute assignment assuming bargaining with n_2 first, and then with n_3. n_1 broadcasts the assignment to both n_2 and n_3. This expands the bargaining group to (n_1, n_2, n_3) without adding extra signalling overhead. However, this also requires that n_1 chooses a sequential bargaining order and gets approval from all the group members on the order before conducting bargaining. If one of neighbors disapproves the request, n_1 needs to perform another request. Hence, for simplicity, we restrict this format to only one-to-one bargaining. Fig. 2 illustrates an example of one-to-one bargaining.

1.b) One-buyer-multi-seller bargaining- A buyer node n_1 purchases a set of channels M_0, from its neighbors who are currently using any channel in M_0, such that to improve system utility. In this case, the bargaining requires concurrent approval from multiple neighbors. As we will show later, this type of bargaining is necessary to eliminate user starvation. Fig. 2 illustrates an example with one buyer and four sellers.

Constraint 2: Self-contained Group Bargaining
Once the bargaining groups are organized, the bargaining inside each group should not disturb the spectrum assignment at nodes outside the group. That is, after the bargaining, the modified channel assignment should not lead to any conflict with nodes outside the group. This helps to maintain system stability, so that a bargaining may not invoke a series of reactions due to violations in interference constraints. More importantly, this guarantees that if a bargaining improves the utility in a local area, it also improves the system utility. Or in other words, a local improvement will lead to a system improvement. This constraint has two components.

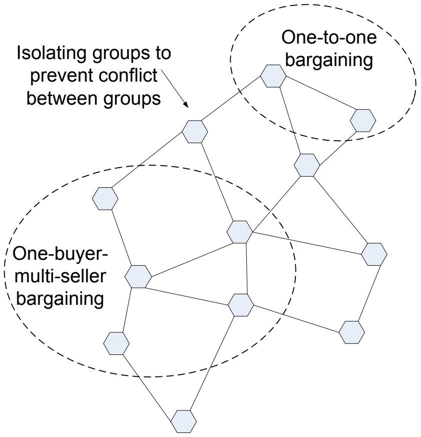


Fig. 2. An example of Bargaining Groups.

2.a) Restricted Bargainable Channels- This restricts the set of channels that are exchangeable between nodes inside each bargaining group, such that when a node gets one channel from its neighbor, the assignment does not conflict with its neighbors outside the bargaining group.

2.b) Isolated Bargaining Group- This not only restricts each node to participate in at most one bargaining group at any time, but also requires that the members of any two bargaining groups can not be directly connected. Having nodes between groups regulate spectrum adjustment and prevents conflict between groups. The necessity of this requirement can be explained by the following example. Assume there are two neighboring nodes A and B (two nodes in the conflict graph connected with an edge) who are the members of two different bargaining groups. Before assignment, A and B are not using channel 0. After the bargaining, both A and B are granted with channel 0 from their bargaining neighbors. However, as A and B conflict with each other if using the same channel, the bargaining produces interference conflict among nodes. The detailed procedure to form isolated bargaining groups will be introduced in Section III-B. An example of isolation between bargaining groups is shown in Fig. 2.

B. Bargaining steps in detail

We design a local bargaining procedure based on the above constraints, assuming a distributed architecture. We propose a *distributed, iterative grouping and bargaining process*. We assume that nodes periodically broadcast their current channel assignment and interference constraints to their neighbors. Each node has three states: *bargaining*, *disabled* and *enabled* (see Fig. 3). Only *enabled* nodes can perform bargaining. The actual bargaining involves the following 4 steps, and repeats until no further bargaining can improve system utility. Here nodes refer to the vertices in the conflict graph, and two "connected" nodes might be physically k $(k > 1$ hops away. Information exchange between them is done through relay.

1.) Initialize Bargaining Request
In general, nodes affected by mobility events initialize bargaining. Based on broadcasts of channel assignments and interference constraints from neighbors, an *enabled* node determines if bargaining with a neighbor will lead to an improvement in system utility. If such neighbors exist, the node broadcasts a bargain request to the neighbors along with its current channel assignment and interference constraints. Such broadcasts reduces communication overhead. As we will show in Section IV-C, additional criterion exists to guide nodes on generating bargaining requests.

2.) Acknowledge Bargaining Request
Neighbors who are *enabled* and willing to bargain reply an *ACK* message with its current channel assignment and interference constraints. We assume that nodes are willing to collaborate to improve system utility, and accept requests that improve system utility even if it might degrade their individual channel assignments. Incentive systems to encourage such collaboration will be investigated in a future paper. If a node receives multiple concurrent requests from its neighbors, it acknowledges the request that leads to the highest bargaining gain as calculated based on information embedded in the request.

3.) Bargain Group Formation
When the requester receives the replies, it selects the members of the bargaining group, and broadcasts this information along with the proposed modification of the channel assignment to neighbors. Once the bargaining group is set, its members enter *bargaining* state. They broadcast a *DISABLE* message with a timer equal to the estimated duration of the bargain process to neighbors not in the bargaining group. Note that the *DISABLE* message can be embedded in the *ACK* messages to reduce overhead. Nodes receiving the message enter *disabled* state for the duration of the timer. This procedure prevents nodes who are neighbors of existing bargaining group to participate in any future bargaining before the timer expires. Following this, all bargaining groups are isolated.

4.) Bargaining
Once all members acknowledge the changes to the channel assignment, each member updates its local channel assignment. This is straightforward for one-to-one bargaining. For one-buyer-multiple-seller bargaining, interactions among members can be coordinated by the bargain requestor. After bargaining, each member enters *enabled* state. Fig. 3 and 4 illustrate the node state transition and messages during bargaining.

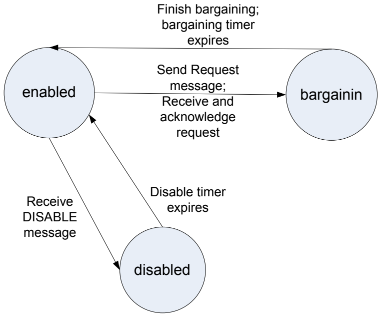

Finish bargaining; bargaining timer expires
enabled
Send Request message; Receive and acknowledge request
bargainin
Receive DISABLE message
Disable timer expires
disabled

Fig. 3. Node State and Transitions

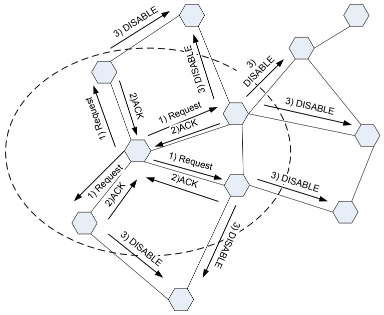


Fig. 4. Messages exchanged during bargaining.

Once the local bargaining procedure is set, the specific bargaining strategy may be customized for different utility functions. It is easy to show that for utilization based utility (total user throughput), the optimization can be reduced to solving M optimization problems for each color respectively. On each color, the corresponding optimization problem is exactly a Weighted Independent Set (WIS) problem [1]. WIS problem is a special case of Weighted Set Packing problem, and can be approximated by a local improvement heuristic algorithm, generally called $t - improvement$ [8], [1]. It is straightforward to convert this algorithm to local bargaining among neighbors. We omit the bargaining procedure due to space constraints, and next focus on the local bargaining strategy for the fairness-based utility.

IV. Local Bargaining to Improve Fairness

In this section, we focus on the local bargaining strategy optimizing for fairness. Based on its definition in (1), the optimization aims to maximize the total logarithmic user throughput, *i.e.* the product of user throughput. Therefore, the global fairness utility increases if nodes with many assigned channels "give" some channels to nodes with few assigned channels. In this section, we start by describing basic one-to-one bargaining where two unbalanced nodes exchange channels to improve the local throughput product. We show that such bargaining is limited by the number of bargainable channels and thus not effective against the node starvation problem. We then develop a special case of one-buyer-multi-seller bargaining, referred to as *Feed Poverty* to eliminate node starvation. We also derive a theoretical lower bound of user throughput using local bargaining under a simplified network configuration.

We first define the following notations.

n: a node n in the conflict graph ($0 \leq n \leq N-1$);

$Nbr(n) = \{v \in V | (n,v) \in E\}$: neighbors of n;

$Nbr(X) = \bigcup_{n \in X} Nbr(n) \setminus X$: neighbors of node set X;

$f_A(n) = \{0 \leq m \leq M-1 | a_{m,n} = 1\}$: the set of channels assigned to node n under current assignment A.

A. One-to-One Fairness Bargaining

As we described, one-to-one bargaining allows two neighboring nodes n_1 and n_2 to exchange channels to improve system utility while complying with conflict constraints from the other neighbors. For n_1 and n_2 to bargain, they need to first obtain the channels that are bargainable to avoid disturbing other neighbors, referred to as $C_b(n_1, n_2)$:

$$C_b(n_1,n_2) = L(n_1) \cap L(n_2) \cap \{\{0..M-1\} \setminus \bigcup_{n \in Nbr(n_1,n_2)} f_A(n)\}.$$

Given $C_b(n_1, n_2)$, we can define the one-to-one bargaining regarding fairness as follows:

Definition 2: For an assignment $A_{M \times N}$, an *One-to-One Fairness Bargaining* finds nodes n_1 and n_2, and their bargaining channel set $C_b(n_1, n_2)$, and modifies $A_{M \times N}$ to $A'_{M \times N}$ related to n_1, n_2 and channels $C_b(n_1, n_2)$, such that

$$TP_{A'}(n_1) \cdot TP_{A'}(n_2) > TP_A(n_1) \cdot TP_A(n_2).$$

The One-to-One Fairness Bargaining increases the product of the bargaining users while other nodes' throughput values remain unaffected. Hence, the system fairness increases with each bargaining. The improvement between each pair of nodes (n_1, n_2) can be calculated as $G(n_1, n_2) = \frac{TP_{A'}(n_1) TP_{A'}(n_2)}{TP_A(n_1) TP_A(n_2)} - 1$. This is used in the bargaining process (in Section III) to determine whether a bargaining can improve system utility.

Given (n_1, n_2), assigning channels to n_1 and n_2 to maximize their throughput product is a difficult task. This is because node throughput depends on all channels (including non-bargainable ones) assigned to a node, and the available bandwidth on a channel differs between nodes. The problem is shown to belong to the class of convex programming problems [22]. When the number of bargainable channels ($|C_b(n_1, n_2)|$) is small (*e.g.* $|C_b(n_1, n_2)| < 10$), exhaustive search may be feasible. Otherwise we need to use approximations based on heuristics such as the one given in [22]: first sort channels in $C_b(n_1, n_2)$ (by channel bandwidth), then use a two-band partition to determine the allocation.

The effectiveness of One-to-One bargaining is constrained by the size of $C_b(n_1, n_2)$. In general, due to heavy interference constraints among neighboring nodes, $C_b(.)$ could be very small. Figure 5 illustrates an example where the conflict graph is a chain topology consisting of three nodes A,B,C. Node B is not assigned with any channel and the system utility is zero. We refer to this as user *starvation*. Node a and b cannot bargain due to the constraint from c (*i.e.* $C_b(a, b) = \emptyset$), while node b and c also cannot bargain due to the constraint from a (*i.e.* $C_b(b, c) = \emptyset$). Hence, the *Fairness Bargaining* is not effective to eliminate user *starvation*.

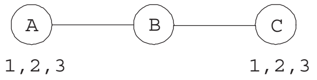


Fig. 5. An example of Starvation

B. Feed Poverty Bargaining

We observe that user *starvation* in most cases is a result of the lack of flexibility in bargaining. As for the example in Figure 5, by allowing A and C to give up channel 1 at the same time and feed it to B, we can remove the starvation at B. This is an example of one-buyer-multi-seller bargaining. In this paper, we propose a special one-buyer-multi-seller bargaining, called *Feed Poverty* where if a node (buyer) has very poor channel assignment, the neighboring nodes can collaborate together to feed it with some channels.

Definition 3: For an assignment $A_{M \times N}$, a *Feed Poverty Bargaining* is to find some node n_0 and channel m_0, modify $A_{M \times N}$ to $A'_{M \times N}$, such that

$$A'_{m,n} = \begin{cases} 1 & : \ m = m_0 \ and \ n = n_0 \\ 0 & : \ m = m_0 \ and \ n \in Nbr(n_0) \\ A_{m,n} & : \ otherwise \end{cases}$$

(intuitively, the assignment let some of n_0's neighbors give up channel m_0 and feed it to n_0) and

$$\begin{aligned} G_{FP}(n_0) &= (TP_{A'}(n_0)) \cdot \prod_{n \in Nbr(n_0) \wedge A_{m_0,n}=1} (TP_{A'}(n)) \\ &- (TP_A(n_0)) \cdot \prod_{n \in Nbr(n_0) \wedge A_{m_0,n}=1} (TP_A(n)) \\ &> 0. \end{aligned}$$

This means the product-throughput of the users involved in the bargaining is locally increasing, while the other users' throughput are not affected. So generally the bargaining improves system utility, except that, in case of starvation of other users, the system utility remains $-\infty$. A special case of Feed Poverty is when $A_{m_0,n} = 0$ for all $n \in Nbr(n_0)$. This means none of n_0's neighbors are using channel m_0, and n_0 simply seizes it.

When there is no feasible *One-to-One Fairness Bargaining*, *i.e.* $|C_b| = \emptyset$, the requestor initializes a *Feed Poverty Bargaining* on all neighbors who acknowledge the request. The requestor sequentially selects multiple channels to maximize group utility.

C. BF-Optimal Assignment and Bound on User Throughput

We propose to combine *one-to-one Fairness Bargaining* and *Feed Poverty Bargaining* into a *Fairness Bargaining with Feed Poverty* (BF). Each node who wants to improve its spectrum usage starts with negotiating *one-to-one Fairness Bargaining* with its neighbors to improve system utility. If there is no bargainable channels between it and any of its neighbors, a *starved* node can broadcast a Feed-Poverty request to its neighbors to initialize *Feed Poverty Bargaining*. Overall, a channel assignment A is said to be *BF-optimal* if no further

Fairness Bargaining with Feed Poverty can be performed on it.

It is useful to derive a theoretical lower bound on each user's throughput for a BF-optimal assignment. However, it is difficult to analyze the system performance when the channel availability and bandwidth vary across channels and users. In the following, we show that when channels are of uniform bandwidth (W.L.O.G. $b_{m,n} = 1$ for all m, n), we can derive the theoretical lower bound on the total number of channels that each user can get, which is equivalent to the user throughput. This also shows an intuition of fairness enforced by *Fairness Bargaining-with-Feed-Poverty*.

Theorem 1: Under a BF-optimal assignment A, for each vertex n in the conflict graph G, $0 \leq n \leq N-1$ with degree $d(n)$ and channel availability list $L(n)$, its spectrum usage $TP(n)$ has a lower bound, *i.e.*

$$TP(n) \geq \left\lfloor \frac{|L(n)|}{d(n)+1} \right\rfloor = PL(n).$$

The proof is included in the appendix. The degree of a vertex $d(n)$ is defined as the number of edges it is associated with, a measure of the number of channel sharers in the neighborhood it has to compete with. Theorem 1 shows that the proposed *Fairness Bargaining with Feed Poverty* guarantees a poverty line $PL(n)$ to each vertex n. The poverty line of a vertex, *i.e.* the throughput a vertex deserves, scales inversely with the number of sharers, which is also the spirit of some greedy allocation algorithms [23], [18]. The poverty line also provides a guideline in bargaining in real systems where a vertex is entitled to request bargaining if its current throughput is below its poverty line. We refer to this as the *Poverty guided bargaining*.

It is easy to show that if channels are fully available at each vertex, *i.e.* $|L(n)| = M$, and the maximum degree in the graph $\Delta = \max_{0 \leq n < N} d(n)$, a BF-optimal assignment can eliminate user starvation if the number of channels $M \geq \Delta + 1$. This matches the well-known conclusion in graph coloring that, the chromatic number of a graph is at most $\Delta + 1$ [5]. It can be shown that the derived poverty bound is also tight, for many typical graph topologies, *e.g.* clique and ring topologies [4].

V. Experimental Results

We conduct experimental simulations to quantify the performance of bargaining-based spectrum allocation. We also validate the proposed local bargaining algorithms against the theoretical lower bounds. For simulations, we assume a noiseless, mobile radio network. We simulate an ad-hoc network by randomly placing a set of nodes on a 100×100 area. We assume that each active node broadcasts data packets to some of its neighbors. We further abstract the network into a *Conflict Graph* where each vertex represents a transmitting node. Any two nodes interfere with each other (*i.e.* connected in the conflict graph) if they are within distance of 20. The actual distance threshold depends on the choice of transmission power and radio hardware. We simply use 20 as an illustrative example. For simplicity, we assume that channels are equally weighted and all the channels are available for each node, *i.e.* $l_{n,m} = 1, \ b_{n,m} = 1$. Our simulations can be easily extended to the cases with partial channel availability and non-uniform channel bandwidth. In terms of traffic demands, all transmitting nodes are assumed backlogged. We focus on maximizing fairness, because bargaining under utility based on spectrum utilization can be reduced to the classical local search of weighted independent set problem, and has been investigated extensively.

Under the simulation setting, for *one-to-one fairness bargaining*, the optimal assignment of channels between two nodes (n_1, n_2) can be derived easily to maximize the product of the number of channels assigned to n_1 and n_2. For *Feed Poverty Bargaining*, we select channel m_0, *i.e.* feeding channel to be the one that generates the minimum disturbance to the neighbors,

$$m_0^* = arg \min_{m_0} \prod_{n \in Nbr(n_0) \wedge A_{m_0,n}=1} \frac{TP_A(n)}{TP_{A'}(n)}.$$

Topology dynamics are modeled by having nodes randomly moving to new locations. We divide time into slots, and in each time slot, $p\%$ of nodes move to a new randomly selected location. The model captures the way mobility is manifested in ad hoc networks without delving into complex protocols. A moving node takes the original channel assignment but disables the channels that conflict with its new neighbors. In each time slot, after the topology change, nodes adjust their channel usage.

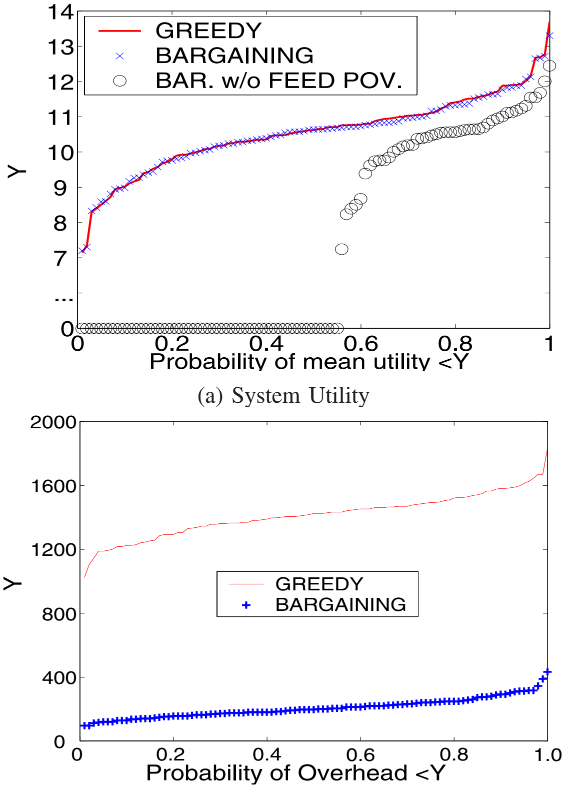
We use two metrics to evaluate the performance.

- System utility: We consider fairness defined in (1). Note that if there exists a user with no channel assigned, the utility becomes $-\infty$. For better representation, we modify the utility to $U(A) = \sqrt[1/N]{\prod_{n=0}^{N-1} TP_A(n)}$ and $U(A) = 0$ if there is any $TP_A(n) = 0$.
- Communication overhead: We quantify algorithm complexity as the communication overhead, *i.e.* total number of messages exchanged among nodes, since transmission and handling of messages will likely dominate computations for channel assignment. In both local bargaining and graph coloring approaches, each iteration of spectrum assignment or bargaining involves a 4-way handshake between neighbors, *i.e.* (request, acknowledgement, action, acknowledgement).

We first compare the performance of local bargaining to the graph coloring approaches that approximate to the solution that maximizes system utility for a given conflict graph [23], [18]. We also validate the impact on system performance when nodes use the derived poverty line to guide its bargaining decision. We then examine the effectiveness of using local bargaining to optimize spectrum assignment for fixed topologies.

A. Comparison with Centralized Graph Coloring Approach

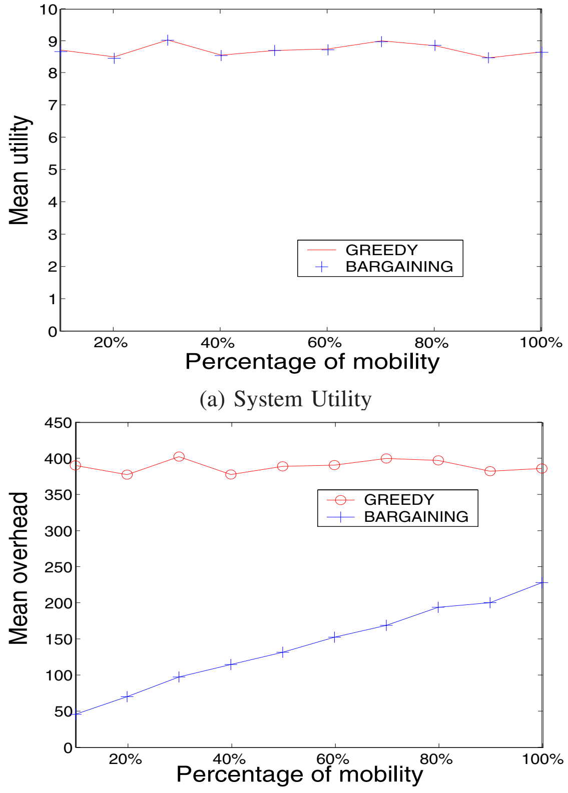
We now compare the proposed local bargaining to the graph-coloring approach. We refer to these two as BARGAINING and GREEDY, respectively. We randomly deploy 40 links



(a) System Utility

(b) Algorithm Complexity

Fig. 6. Performance comparison of GREEDY and BARGAINING



(a) System Utility

(b) Algorithm Complexity

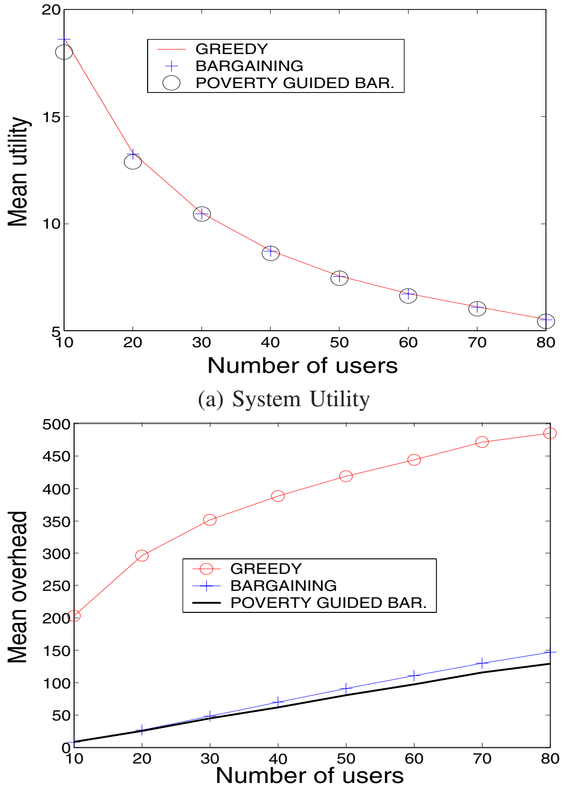
Fig. 7. Performance comparison of GREEDY and BARGAINING under different user mobility.

with 30 channels in a given area and produce the corresponding conflict graph. We use the graph coloring approach to derive an initial spectrum assignment for the given conflict graph. We simulate mobility events in the next 100 time slots, one event per time slot where up to 6 nodes move to new locations. After each event, we apply both local bargaining and graph coloring approaches to derive the new spectrum allocation.

Figure 6(a) illustrates the sorted fairness utility using both approaches, and local bargaining performs nearly as good as the graph coloring approach. The graph coloring approach makes decisions with the knowledge of global topology, while using local bargaining, each user makes decisions based on only neighbor information. Figure 6(b) compares the communication overhead in each time slot. We observe that local bargaining achieves similar performance while incurring much lower complexity in terms of messages exchanged. This significant overhead reduction allows quick adaptation to network dynamics. In Figure 6(a), we also examine the performance of local bargaining using only *one-to-one fairness bargaining*, without *Feed Poverty Bargaining*. The results confirm that *Feed Poverty Bargaining* is required to effectively eliminate user *starvation*.

Next, we extend the simulation to allow $p\%$ of vertices move to new randomly selected locations. In general, larger p implies more disturbance to the conflict graph and thus more vertices will perform bargaining to adapt their spectrum usage to the new topology. Figure 7 illustrates the system utility and algorithm overhead for local bargaining and graph coloring for increasing values of p. The utility is geometrically averaged over 100 time slots, and overhead is averaged over 100 time slots. As before, local bargaining performs similarly to graph coloring approach in system utility. The overhead of graph coloring is not sensitive to the value of p as it mainly depends on the size of the graph in number of vertices and channels. The overhead complexity of local bargaining increases with p as more vertices need to perform local bargaining. We observe that even under 100% mobility, local bargaining results roughly $1/2$ the overhead in terms of messages exchanged compared to the graph coloring approach. Therefore, local bargaining appears to be an attractive alternative to graph coloring for optimizing spectrum allocation on a given network topology.

In Figure 8, we fix the number of channels at 40 and examine the impact of the number of vertices for $p = 20\%$ mobility. Increasing the density of vertices in a fixed area creates additional interference constraints and thus increases average vertex degree in the conflict graph. Therefore, system utility scales inversely with the number of vertices while the algorithm complexity increases. As before, results show that local bargaining compares favorably with graph coloring in quality of allocation while incurring significantly less over-



(a) System Utility

(b) Algorithm Complexity

Fig. 8. Performance comparison of GREEDY and BARGAINING under different vertex density.

head.

B. *Tightness of the Poverty bound*

We now examine the appropriateness of the user poverty bound derived in Theorem 1. Figure 9 illustrates the histogram of the ratio of the actual user throughput and the poverty bound assuming 40 vertices and 100 time slots. Results show that the theoretical bound is valid and fairly tight. As we described, nodes can use the poverty line to decide whether further local bargaining is necessary. A vertex with an assignment below the poverty line should bargain with additional neighbors to acquire additional channels. In addition, if a vertex predicts that a bargaining request from a neighbor will drop its assignment below the poverty line, it can reject the request. In Figure 8, we also compare the performance of *Poverty guided bargaining* to the graph coloring and bargaining approaches. We show that *Poverty guided bargaining* performs close to that of the bargaining but with 10% less overhead, again demonstrating the tightness of the poverty line bound.

C. *Use local bargaining to optimize for a given topology*

As stated before, it is possible to use local bargaining to approximate the graph coloring approach and derive the spectrum allocation for a given topology. Local bargaining starts from a random allocation and gradually improve the system utility. Figure 10 compares the system utility and algorithm complexity using graph coloring (GREEDY), local bargaining (BARGAINING) and random assignment (RANDOM). User starvation is common when using random assignment, resulting in many zero values for system utility. Local bargaining can effectively eliminate user starvation and performs only slightly worse compared to graph coloring approach. Figure 10(b) shows that local bargaining can significantly reduce communication overhead.

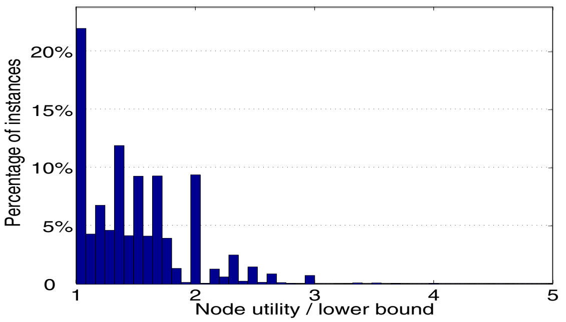
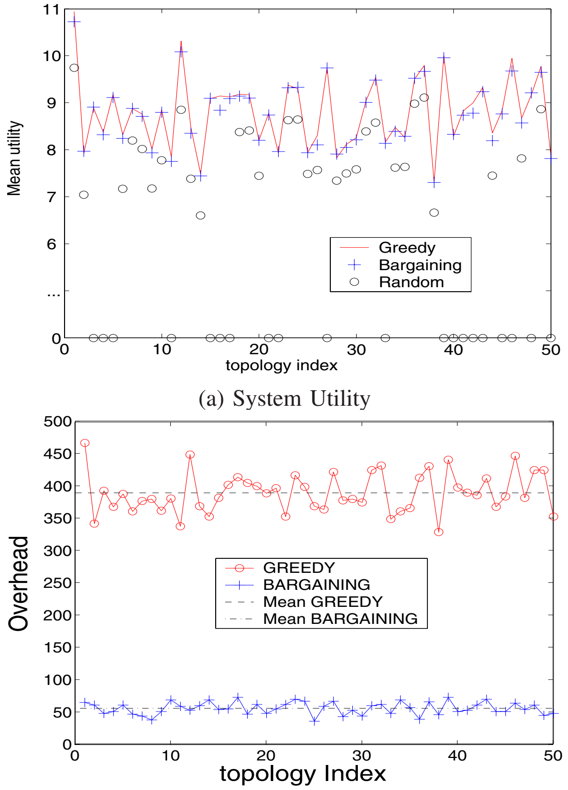


Fig. 9. Histogram of the ratio between the actual user throughput and the poverty bound.



(a) System Utility

(b) Algorithm Complexity

Fig. 10. Performance comparison of graph coloring, local bargaining and random spectrum assignment for static environments.

VI. Theoretical Distance to Social Optimal

In this section, we analyze the performance of the proposed bargaining strategy, by comparing it to that of the socially optimal assignment. A socially optimal assignment is one maximizes global system utility. We are interested in the ratio of the utility of social optimum to that of our bargaining optimum, often referred to as the *price of anarchy (POA)* as defined in [14]. The analysis requires a number of case study and illustrations. We only give a sketch here, and the details will be included in a future paper [4].

It can be shown that the POA is unbounded in general, when there is no restriction on channel bandwidth or number of channels [4]. In the following we assume that channels are of uniform bandwidth (W.L.O.G. $b_{m,n} = 1$ for all m, n), and the number of nodes are bounded. In this case the POA can be bounded, based on the lower bound derived in Section III.

Theorem 2: For a topology with uniform channel availability and channel bandwidth, *i.e.* $b_{m,n} = 1$, $l_{m,n} = 1$, and $M > \Delta$[1], the price of anarchy for a BF-optimal channel assignment is at most

$$\frac{M^N \cdot \prod_{\langle u,v\rangle} \left(\frac{d_v}{d_u+d_v}\right)^{\frac{1}{d_u}} \cdot \left(\frac{d_u}{d_u+d_v}\right)^{\frac{1}{d_v}}}{\prod_{n=0}^{N-1} \left\lfloor \frac{M}{d_n+1} \right\rfloor}. \tag{2}$$

The proof is in the appendix.

VII. Related Work

Optimal conflict-free channel assignment satisfying a global optimal objective is often NP-hard, even when global topology information is available [6]. Centralized approximations are widely used in single hop wireless networks such as cellular networks. This can be easily extended to multi-hop wireless networks by flooding connectivity and traffic requirements across the network, and requiring all users to run a variant of the centralized algorithm. However, this approach clearly does not scale as networks become larger and more dynamic.

An alternative decentralized allocation, where users act based on locally available information is much more attractive. Both analytical framework and practical strategies have been proposed. Analytical frameworks in [17], [12] address fairness for single-hop flows, and derive an estimate of the rate at each flow to achieve Max-Min fairness. However, there is no guarantee that a feasible scheme exists to achieve the rate.

Practical strategies have been proposed for sharing a single channel. Contention based schemes invoke a random access protocol like ALOHA and CSMA, where users contend in time to share a common channel [15], [12], [17]. While this scheme provides fairness and utilization on a single channel system probabilistically, its application to a multi-channel system requires each user to know how many and which channel(s) to access. Another approach, conflict free time slot scheduling, provides guaranteed channel usage by reserving time slots for each flow. Solutions in [20], [2], [19] assign exactly one time slot to each flow. This approach can be used in multi-channel systems if each user uses only one channel. Another solution [21] allows users to use multiple slots/channels to achieve Max-Min-fair, but does not consider interference from neighbor transmissions.

Multi-channel assignment strategies were developed mostly for cellular networks. The work in [13] provides solutions to assign frequency bands among base stations to minimize call blocking probability for voice traffic. There is no notion of fairness as the traffic determines the number of channels each base station should use. In [9], the authors proposed a graph-theoretic model and discussed the price of anarchy under various topology conditions such as different channel numbers and bargaining strategies. The main difference between [9] and the proposed work is that the proposed model allows multi-coloring of a vertex, while in [9] each vertex can only be assigned with at most one color.

In [23], [18], the authors presented a generalized spectrum allocation problem where interference constraint C is channel dependent. The authors developed a set of greedy coloring approach to optimize spectrum allocation for a given conflict graph. We use the proposed approaches in [23], [18] as the reference algorithm, *i.e. GREEDY* in this paper.

Cooperative/non-cooperative bargaining is also used in previous research to optimize channel allocation for cellular networks. In [10] and [11], the authors proposed a set of bargaining strategies for OFDMA based network, focusing on one-to-one bargaining. In these cases, nodes are mutually interfered, *i.e.* the corresponding conflict graph is fully-connected. Forming bargaining group is to find any two users network and let them exchange certain channels to improve system performance. The main difference between these work and the proposed work is that the proposed work provides solutions for general conflict graphs where group setup needs to consider local topology (*i.e.* isolated group and self-contained channel adjustment). We propose a feed-poverty bargaining to eliminate user starvation which can not addressed by one-to-one bargaining. In addition, the proposed work derives a lower bound for each node's channel assignment that does not limit to fully-connected topology.

VIII. Conclusion and Future Work

In this paper, we present an adaptive and distributed approach to spectrum allocation in mobile ad-hoc networks. We propose a local bargaining approach where users affected by the mobility event self-organize into bargaining groups and adapt their spectrum assignment to approximate a new optimal assignment. The bargaining starts from the spectrum assignment retained before the mobility event, and requires significantly less computation and communication overhead. We propose a *Fairness Bargaining with Feed Poverty* to improve fairness in spectrum assignment. We also derive a lower bound on the spectrum assignment that each node can get from bargaining, referred to as the *poverty line*. It reflects the level of fairness enforced by the bargaining, and serves as a guideline to organize bargaining. Experimental results show that the proposed bargaining approach performs similarly as the topology-optimized approach but with much less complexity. We also verify the correctness of the *poverty line* and the effectiveness of the *poverty guided bargaining*.

[1] otherwise there may exist starvation and the ratio may be meaningless.

While we only proposed a specific bargaining strategy to maximize fairness based system utility, the proposed bargaining framework can be extended towards other utility functions or optimization goal. We intend to examine this in a future study. One of the biggest attractions of local bargaining is the low complexity. We have conducted simulations to evaluate algorithm complexity in terms of the number of bargaining iterations, which is quite intuitive. We are currently working on a theoretical analysis on the complexity of local bargaining. In addition, the proposed bargaining framework assumes that network nodes collaborate to improve system utility while in real systems, nodes can be selfish so that a pricing based bargaining or a rule based bargaining would be more practical. We are currently investigating this issue.

REFERENCES

[1] ARKIN, E. M., AND HASSIN, R. On local search for weighted k-set packing. In *proceedings of ESA, LNCS 1284* (1997), pp. 13–22.

[2] BAO, L., AND GARCIA-LUNA-ACEVES, J. J. Hybrid channel access scheduling in ad hoc networks. In *Prof. ICNP* (2002).

[3] BERGER, R. J. Open spectrum: a path to ubiquitous connectivity. *ACM Queue 1*, 3 (May 2003).

[4] CAO, L., AND ZHENG, H. A game theory approach towards fair spectrum sharing. work in process.

[5] CHARTRAND, G. *A Scheduling Problem: An Introduction to Chromatic Numbers*. New York: Dover, pp.202-209, 1985.

[6] EPHREMIDES, A., AND TRUONG, T. Scheduling broadcasts in multihop radio networks. In *IEEE Trans. on Communications* (April 1990), vol. 38, pp. 456–460.

[7] GAREY, M. R., AND JOHNSON, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1990.

[8] HALLDÓRSSON, M. M. Approximations of independent sets in graphs. In *proceedings of APPROX '98: Proceedings of the International Workshop on Approximation Algorithms for Combinatorial Optimization, LNCS 1444* (1998), Springer-Verlag, pp. 1–13.

[9] HALLDÓRSSON, M. M., HALPERN, J. Y., LI, L. E., AND MIRROKNI, V. S. On spectrum sharing games. In *PODC '04: Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing* (2004), ACM Press, pp. 107–114.

[10] HAN, Z., JI, Z., AND LIU, K. R. Low-complexity OFDMA channel allocation with Nash bargaining solution fairness. In *IEEE GLOBECOM '04* (2004).

[11] HAN, Z., JI, Z., AND LIU, K. R. Power minimization for multi-cell OFDM networks using distributed non-cooperative game approach. In *IEEE GLOBECOM '04* (2004).

[12] HUANG, X., AND B.BENSAOU. On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation. In *Proc. of Mobihoc* (2001), ACM.

[13] KATZELA, I., AND NAGHSHINEH, M. Channel assignment schems for celluar mobile telecommunication systems. *IEEE Personal Communications 3*, 3 (June 1996), 10–31.

[14] KOUTSOUPIAS, E., AND PAPADIMITRIOU, C. Worst-case equilibria. In *Proc. 16th Annual Conf. Theoretical Aspects of Computer Science* (1999), vol. 1563 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 404–413.

[15] LUO, H., LU, S., AND BHARGHAVAN, V. A new model for packet scheduling in multihop wireless networks. In *Proc. ACM Mobicom* (August 2000).

[16] MCHENRY, M. Spectrum white space measurements. *New America Foundation Broadband Forum* (June 2003).

[17] NANDAGOPAL, T., T.KIM, X.GAO, AND BHARGHAVAN, V. Achieving mac layer fairness in wireless packet networks. In *Proc. ACM Mobicom* (August 2000).

[18] PENG, C., ZHENG, H., AND ZHAO, B. Y. Utilization and fairness in spectrum assignemnt for opportunistic spectrum access. *Mobile Networks and Applications (MONET)* (2006). to appear.

[19] RAMANATHAN, S. A unified framework and algorithm for channel assignment in wireless networks. *Wireless Networks 5*, 2 (March 1999), 81–94.

[20] RAMANATHAN, S., AND LLOYD, E. Scheduling algorithms for multihop radio networks. In *IEEE/ACM Trans. on Networking* (April 1993), vol. 1, pp. 166–177.

[21] SALONIDIS, T., AND TASSIULAS, L. Distributed on-line schedule adapation for balanced slot allocation in wireless ad hoc networks. In *Proc. of IWQoS* (2004).

[22] YU, W., AND CIOFFI, J. FDMA capacity of gaussian multi-access channels with ISI. *IEEE Trans. Commun. 50*, 1 (Jan 2002), 102–111.

[23] ZHENG, H., AND PENG, C. Collaboration and fairness in opportunistic spectrum access. In *Proc. 40th annual IEEE International Conference on Communications* (June 2005).

APPENDIX

A. *Proof of Theorem 1*

The proof for Theorem 1 is complex. We start by proving the following corollary, which is a special case of the Theorem. We then provide a sketch of generalization to Theorem 1.

Corollary 1: Under a BF-optimal assignment A, for each vertex in the conflict graph G, with degree $d(n)$ and channel availability list $L(n) = \{0, 1, \cdots, M-1\}$, the following holds:

$$TP(n) \geq \left\lfloor \frac{M}{d(n)+1} \right\rfloor, \ 0 \leq n \leq N-1. \tag{3}$$

This is the case where each vertex has full channel availability.

A.1 *Proof of Corollary 1*

We first show the following lemma exists.

Lemma 1: Let $B_{P\times Q}$ represent a $P \times Q$ binary matrix. Define $TP(n) = \sum_{m=0}^{P-1} B_{m,n}$, for $0 \leq n \leq Q-1$; and $L_c = \prod_{0\leq i\leq Q-1, B_{c,i}=1} \frac{TP(i)-1}{TP(i)}$. There exists $0 \leq c \leq P-1$, s.t.

$$L_c \geq \frac{r-1}{r}, \ \ r = P/Q. \tag{4}$$

Proof of Lemma 1.

Proof: The proof is trivial when $0 < r < 1$ since the right side is negative. Next, we assume $r \geq 1$ and use an induction on Q to prove the lemma.

When $Q = 1$, $r = P$, B becomes a $r \times 1$ binary matrix. If for some $0 \leq c \leq r-1$, $B_{c,0} = 0$, then c satisfies (4). Otherwise, if $B_{c,0} = 1$ $\forall 0 \leq c \leq r-1$, then $TP(0) = r$, and $\forall c$, $L_c = \frac{r-1}{r}$.

Now suppose the lemma holds for all $k < Q$ $(Q \geq 2)$.

a) We first assume that there exists $0 \leq n_0 \leq Q-1$, s.t. $TP(n_0) \leq \lfloor r \rfloor$, *e.g.* $n_0 = 0$. Hence, $\forall j$, $\lfloor r \rfloor \leq j \leq P-1$, $B_{j,0} = 0$. Then, $B_{(\lfloor r \rfloor : P-1),(1:Q-1)}$ is a binary matrix constructed by deleting the first $\lfloor r \rfloor$ rows and the first column of B. Its size is $(P - \lfloor r \rfloor) \times (Q-1)$. By induction hypothesis, there exists $\lfloor r \rfloor \leq c \leq P-1$, *s.t.*

$$\prod_{1\leq i\leq Q-1, B_{c,i}=1} \frac{(\sum_{m=\lfloor r \rfloor}^{P-1} B_{m,i}) - 1}{(\sum_{m=\lfloor r \rfloor}^{P-1} B_{m,i})} \geq \frac{r'-1}{r'}, \tag{5}$$

where $r' = \frac{P-\lfloor r \rfloor}{Q-1} \geq \frac{P-r}{Q-1} = r$. Following this, we can derive L_c as

$$\begin{aligned} L_c &= \prod_{0\leq i\leq Q-1, B_{c,i}=1} \frac{TP(i)-1}{TP(i)} \\ &= \prod_{0\leq i\leq Q-1, B_{c,i}=1} \frac{(\sum_{m=0}^{P-1} B_{m,i}) - 1}{\sum_{m=0}^{P-1} B_{m,i}} \\ (B_{c,0} = 0) \quad &= \prod_{1\leq i\leq Q-1, B_{c,i}=1} \frac{(\sum_{m=0}^{P-1} B_{m,i}) - 1}{\sum_{m=0}^{P-1} B_{m,i}} \\ &\geq \prod_{1\leq i\leq Q-1, B_{c,i}=1} \frac{(\sum_{m=\lfloor r \rfloor}^{P-1} B_{m,i}) - 1}{\sum_{m=\lfloor r \rfloor}^{P-1} B_{m,i}} \\ (by (5)) \quad &\geq \frac{r'-1}{r'} \geq \frac{r-1}{r}. \end{aligned} \tag{6}$$

b) We now assume that $\forall n,\ 0 \le n \le Q-1, TP(n) > \lfloor r \rfloor$. Since $TP(n)$ is an integer, $TP(n) > r$. Accordingly,

$$\begin{aligned}
\prod_{c=0}^{P-1} L_c &= \prod_{0\le c\le P-1} \prod_{0\le i\le Q-1, B_{c,i}=1} \frac{TP(i)-1}{TP(i)} \\
&= \prod_{0\le i\le Q-1} \prod_{0\le c\le P-1, B_{c,i}=1} \frac{TP(i)-1}{TP(i)} \\
&= \prod_{0\le i\le Q-1} \left(\frac{TP(i)-1}{TP(i)}\right)^{TP(i)} \\
(Lemma 2) \quad &\ge \prod_{0\le i\le Q-1} \left(\frac{r-1}{r}\right)^{r} \\
&= \left(\frac{r-1}{r}\right)^{r\cdot Q} = \left(\frac{r-1}{r}\right)^{P}. \qquad (7)
\end{aligned}$$

From the above, we show that $\prod_{c=0}^{P-1} L_c \ge (\frac{r-1}{r})^P$. Because $L_c \ge 0$ for all $0 \le c \le P-1$, there must exist $0 \le c \le P-1$, s.t. $L_c \ge \frac{r-1}{r}$. ■

The following lemma is also required to prove the corollary. Its proof is straightforward and omitted due to space limit.

Lemma 2: Let $f(x) = (1-\frac{1}{x})^x$. Then $f(x)$ is monotonically increasing in $[1, +\infty)$.

Proof of Corollary 1.
Proof: Define $r = \lfloor \frac{M}{d+1} \rfloor$, and we have $M = (d+1)\cdot r + r_0,\ r, r_0 \in \mathcal{Z}, 0 \le r_0 \le d$. If $r = 0$, the corollary is trivial. In the following we assume $r > 0$. We will show that $\nexists n,\ TP(n) < \left\lfloor \frac{M}{d+1} \right\rfloor$.
If $\exists n, TP(n) < \left\lfloor \frac{M}{d+1} \right\rfloor$, then

$$TP(n) \le \left\lfloor \frac{M}{d+1} \right\rfloor - 1 = r - 1. \qquad (8)$$

We assume that user n has d neighbors indexed $\{0, 1, \cdots, d-1\}$. We will use d to index user n. We assume that the channels assigned to user n, $f_A(n)$are indexed by $\{M-1, M-2, \cdots, M-TP(n)\}$. The assignment matrix can be illustrated by,

Index	0	$\cdots$	d-1	d	$\cdots$
0	$\cdots$	$\cdots$	$\cdots$	0	$\cdots$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\cdots$	$\vdots$
$M-TP(n)-1$	$\cdots$	$\cdots$	$\cdots$	0	$\cdots$
$M-TP(n)$	0	$\cdots$	0	1	$\cdots$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
M-1	0	$\cdots$	0	1	$\cdots$

Now $A_{(0:M-TP(n)-1),(0:d-1)}$ satisfies the conditions in Lemma 1, with $P = M - TP(n)$ and $Q = d$. By Lemma 1, there exists $0 \le c \le M - TP(n) - 1$, s.t.

$$L_c = \prod_{0\le i\le d-1, A_{c,i}=1} \frac{(\sum_{m=0}^{M-TP(n)-1} A_{m,i}) - 1}{(\sum_{m=0}^{M-TP(n)-1} A_{m,i})} \ge \frac{r'-1}{r'}. \qquad (9)$$

Here

$$\begin{aligned}
r' &= \frac{M-TP(n)}{d} \ge \frac{M-(r-1)}{d} \\
&= \frac{((d+1)\cdot r + r_0) - (r-1)}{d} > r. \qquad (10)
\end{aligned}$$

Since $A_{m,i} = 0$ for $M-TP(n) \le m \le M-1$ and $0 \le i \le d-1$, it is easy to show that $\sum_{m=0}^{M-TP(n)-1} A_{m,i} = \sum_{m=0}^{M-1} A_{m,i} = TP(i)$, for $0 \le i \le d-1$.

Based on (9), we can derive the following,

$$\prod_{0\le i\le d-1, A_{c,i}=1} \frac{TP(i)-1}{TP(i)} \ge \frac{r'-1}{r'} > \frac{r-1}{r}. \qquad (11)$$

Now there are two cases:
- $TP(n) = 0$. Then

$$\begin{aligned}
&(TP(n)+1) \cdot \prod_{i\in Nbr(n)\wedge A_{c,i}=1} (TP(i)-1) \\
&> 0\ (by (11)) \\
&= (TP(n)) \cdot \prod_{i\in Nbr(n)\wedge A_{c,i}=1} (TP(i)). \qquad (12)
\end{aligned}$$

- $TP(n) > 0$. Then

$$\begin{aligned}
&\frac{(TP(n)+1) \cdot \prod_{i\in Nbr(n)\wedge A_{c,i}=1}(TP(i)-1)}{(TP(n)) \cdot \prod_{i\in Nbr(n)\wedge A_{c,i}=1}(TP(i))} \\
&= \frac{TP(n)+1}{TP(n)} \cdot \left(\prod_{0\le i\le d-1, A_{c,i}=1} \frac{TP(i)-1}{TP(i)}\right) \\
&(by (11)) > \frac{TP(n)+1}{TP(n)} \cdot \frac{r-1}{r} \\
&(by (8)) \ \ \ge \frac{(r-1)+1}{r-1} \cdot \frac{r-1}{r} = 1. \qquad (13)
\end{aligned}$$

(12) and (13) show that we can apply *Feed Poverty* on assignment A, by feeding channel c to node n. This contradicts with the assumption that assignment A is BF-optimal. ■

A.2 *Generalize to Theorem 1.*
We can prove Theorem 1 using a similar argument as Corollary 1. Suppose there is a node n who violates the inequality, then there must exist a channel $c \in L(n)$, s.t. when c is feeded to n and removed from its neighbors, the system utility increases. This is done by considering the sub assignment matrix $A_{L(n)\times(0:N-1)}$ instead of $A_{(0:M-1)\times(0:N-1)}$ in the proof [4].

B. *Proof of Theorem 2*

The proof can be reduced to deriving a upper bound of social optimum.
Proof: Based on the observation that for each pair $\langle u, v\rangle$ of neighboring nodes, $TP(u) + TP(v) \le M$. We have

$$\begin{aligned}
&TP(u)^{\frac{1}{d_u}} \cdot TP(v)^{\frac{1}{d_v}} \\
\le\ &\left(\frac{d_v}{d_u+d_v}\cdot M\right)^{\frac{1}{d_u}} \cdot \left(\frac{d_u}{d_u+d_v}\cdot M\right)^{\frac{1}{d_v}}.
\end{aligned}$$

Thus,

$$\begin{aligned}
&\prod_{n=0}^{N-1} TP(n) = \prod_{\langle u,v\rangle} TP(u)^{\frac{1}{d_u}} \cdot TP(v)^{\frac{1}{d_v}} \\
\le\ &\prod_{\langle u,v\rangle} \left(\frac{d_v}{d_u+d_v}\cdot M\right)^{\frac{1}{d_u}} \cdot \left(\frac{d_u}{d_u+d_v}\cdot M\right)^{\frac{1}{d_v}} \\
=\ &M^N \cdot \prod_{\langle u,v\rangle} \left(\frac{d_v}{d_u+d_v}\right)^{\frac{1}{d_u}} \cdot \left(\frac{d_u}{d_u+d_v}\right)^{\frac{1}{d_v}}. \qquad (14)
\end{aligned}$$

Therefore, based on Theorem 1 and (14) , the price of anarchy is at most

$$\frac{M^N \cdot \prod_{\langle u,v\rangle} \left(\frac{d_v}{d_u+d_v}\right)^{\frac{1}{d_u}} \cdot \left(\frac{d_u}{d_u+d_v}\right)^{\frac{1}{d_v}}}{\prod_{n=0}^{N-1} \left\lfloor \frac{M}{d_n+1} \right\rfloor}. \qquad (15)$$

■