Site-Specific Knowledge and Interference Measurement for Improving Frequency Allocations in Wireless Networks

Jeremy K. Chen, *Member, IEEE*, Gustavo de Veciana, *Fellow, IEEE*, and Theodore S. Rappaport, *Fellow, IEEE*

***Abstract*—We present new frequency allocation schemes for wireless networks and show that they outperform all other published work. Two categories of schemes are presented: 1) those purely based on measurements and 2) those that use site-specific knowledge, which refers to knowledge of building layouts, the locations and electrical properties of access points (APs), users, and physical objects. In our site-specific knowledge-based algorithms, a central network controller communicates with all APs and has site-specific knowledge so that it can *a priori* predict the received power from any transmitter to any receiver. Optimal frequency assignments are based on predicted powers to minimize interference and maximize throughput. In our measurement-based algorithms, clients periodically report *in situ* interference measurements to their associated APs; then, the APs' frequency allocations are adjusted based on the reported measurements. Unlike other work, we minimize interference seen by both *users* and *APs*, use a *physical* model rather than a *binary* model for interference, and mitigate the impact of rogue interference. Our algorithms consistently yield high throughput gains, irrespective of the network topology, AP activity level, number of APs, rogue interferers, and available channels. Our algorithms outperform the best published work by 18.5%, 97.6%, and 1180% for median, 25th percentile, and 15th percentile user throughputs, respectively.**

***Index Terms*—Cellular networks, frequency allocation, radio spectrum management, site-specific knowledge, wireless local area network (WLAN).**

I. Introduction

RADIO propagation characteristics are fundamentally site specific, since radio propagation mechanisms (e.g., penetration, reflection, and diffraction) are directly related to the locations, sizes, and electrical properties of physical objects in the surroundings. Site-specific channel prediction algorithms are well understood [3]–[7]. These site-specific prediction techniques use a building layout or a satellite map and compute path losses between any two locations in indoor or outdoor environments. The complexity of these prediction tools has been reduced, and computing power has increased, so that they can be implemented for real-time network management applications. This paper is the first work to analyze how site-specific knowledge can improve ongoing frequency allocation in wireless local area networks (WLANs). In WLANs, a number of orthogonal frequency channels are allocated, and each AP is allocated one channel. When the number of channels is limited relative to the number of APs, some APs inevitably use the same channel and induce cochannel interference. The same problem exists in cellular networks. Judicious channel reuse mechanisms are necessary to reduce interference, particularly for the case of mobile users, such as in enterprise voice-over-IP networks or in cellular networks. Today, site-specific knowledge is becoming much more available from blueprints, AutoCAD, and Google Maps and Google Earth, for example. This paper demonstrates vast improvement in overall network performance, particularly by greatly raising the throughputs of the users that suffer low throughputs. In other words, our schemes enable many more users to have acceptable throughputs for file transfer or voice-over-IP applications, as compared to prior published work.

A number of WLAN frequency allocation schemes have been proposed thus far. The work in [8] assumes each AP has a different fixed traffic load and defines the effective channel utilization of an AP as the fraction of time the channel is used for data transmission or is sensed busy due to interference from other APs; then, the maximum effective channel utilization among all APs is minimized. AP placement and frequency allocation are jointly optimized in [9] with the same objective of minimizing the max channel utilization as in [8]. The work in [8] and [9] fails to maximize throughput or minimize interference seen by clients and, hence, perform poorer than our proposed work in today's WLANs, where downlink traffic dominates. The frequency allocation problem is modeled as a minimum-sum-weight vertex-coloring problem in [10], where vertices are APs, and the weight of each edge between two APs denotes the number of clients that are associated with either one of these two APs and are interfered by the other AP. The work in [11] minimizes the number of clients whose transmissions suffer channel conflicts; a client associated with an AP suffers conflicts if other clients or other APs interfere with the client or the AP under consideration. The definition of channel conflict in [11] is more comprehensive than those in [8]–[10]. The work in [11] has been shown to outperform

Manuscript received July 26, 2008. First published December 12, 2008; current version published May 11, 2009. This work was supported by the National Science Foundation under Grant ACI-0305644 and Grant CNS-0325788. This paper was presented in part at the IEEE Vehicular Technology Conference, Baltimore MD, October 2007 [1] and the IEEE Global Communications Conference, Washington, DC, November 2007 [2]. The review of this paper was coordinated by Dr. K. Zangi.

J. K. Chen was with the Wireless Networking and Communications Group, The University of Texas at Austin, Austin, TX 78712 USA. He is now with Google Inc., New York, NY 10011 USA (e-mail: jeremychen@google.com).

G. de Veciana and T. S. Rappaport are with the Wireless Networking and Communications Group, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: gustavo@ece.utexas.edu; wireless@ece.utexas.edu).

Digital Object Identifier 10.1109/TVT.2008.2010862

[8]–[10] but performs poorer than our proposed schemes in the presence of *rogue interferers*, i.e., intentional or unintentional RF interferers, microwave ovens, or other RF devices that also operate on the same unlicensed bands as WLANs.

Only [12] presents mechanisms to detect and reduce the negative impact from rogue interferers. In [12], each AP senses interference and independently selects a channel whose measured interference power is below a predefined threshold, without coordinating with other APs. In networks with high interference, it may not be feasible to find a channel allocation so that every AP senses interference below the threshold; in this case, the algorithm in [12] does not converge. One could in principle set a higher threshold for the algorithm in [12] to work in high-interference regimes, but [12] does not mention methods to adapt the threshold. It is not trivial to adapt this threshold, since a high threshold will degrade network performance, but a low threshold will yield no feasible solution. By contrast, four of our proposed algorithms converge irrespective of the overall interference level. The nonconvergence result of [12] in the high-interference regime is due to the *binary model for interference*, which is also used in [8]–[11]. Our work considers a *physical* model for interference; that is, we assume that interference power is a continuous quantity, which properly represents the real world. Therefore, our work performs better than [12].

Most traffic in WLANs is downlink [6], [13]; hence, maximizing *downlink* throughputs and signal-to-interference-and-noise ratios (SINRs) seen by *users* are key to proper network design. The work in [8], [9], and [12] focuses on minimizing the interference at *APs* rather than that seen by *users*, as is done in [10] and [11], and thus often perform poorer than [10] and [11].

A. Main Contribution

The *main contribution* of this paper is our two categories of new algorithms for channel allocations that outperform all other published work, i.e., those in [8]–[12]. In the remainder of this paper, we mainly present the gains of our proposed algorithms against the works in [11] and [12], since the work in [11] has been shown to outperform [8]–[10]. The proposed algorithms perform well mainly because of the following reasons: 1) They minimize interference seen by *users* rather than that seen by *APs*; 2) they use a *physical* model rather than a *binary* model for interference; and 3) they have the ability to deal with *rogue interferers*. The first category is based on interference measurements at APs and users, whereas the second category is based on site-specific knowledge. We describe our ideas and contribution in the following discussion.

1) Measurement-Based Algorithms: We propose that all or a subset of clients periodically measure the *in situ* interference power on all frequency channels when their associated APs are idle, and report the average measured power to their associated APs. This technique is used in mobile-assisted handoff (MAHO) in the cellular field [14], and results in this paper may also be applied to cellular networks. APs also measure *in situ* interference power. Since the measurements at APs or clients are performed during their idle time, the overhead is negligible. Each AP then computes a metric, called *weighted interference*, which captures the overall interference as seen by itself and its clients by placing different weights on its and the clients' *in situ* measurements according to the clients' traffic loads, signal strengths, and uplink and downlink traffic volume. Section V shows by simulation that our measurement-based algorithms substantially outperform [11] and [12]. Since the work in [11] has been shown to outperform [8]–[10], we conclude that our algorithms outperform [8]–[12].

2) Site-Specific Knowledge-Based Algorithms: The aforementioned measurement-based algorithms can still be improved if we assume that a central network controller has and uses site-specific knowledge to optimize frequency allocation of each AP and each user. The advantage of using site-specific knowledge is to *a priori* predict the path loss between any pair of AP and user when the user's location is obtained via the Global Positioning System (GPS) or other known position location technologies.[1] The predicted path losses can help formulate a global optimization problem, thereby maximizing throughputs and saving power, etc. Although the environment affects the path losses, empirical results show that by modeling large fixed partitions and items in the environment (such as walls, book shelves, and cubicles), the *predicted* and the *measured* path losses show high agreement (e.g., the mean error is less than 4 dB) [3]–[6].

Distributed measurement-based algorithms with the knowledge of APs' transmit powers via message exchanges can learn over time the path loss or received power between every transmitter and every receiver; examples of such algorithms are described in [17]. Nevertheless, the time needed for learning may be too long when the number of interfering APs is large. The overall learning time could be shortened if each client learns the interference power from only the APs that are in the range of causing nonnegligible interference at the client. To know which base stations are in the interference range, site-specific knowledge (such as the environments and the locations of APs and clients) is needed. Saving the learning time for measurement-based algorithms is a topic for ongoing and future work. In this paper, we choose to use site-specific knowledge to *a priori* predict the interference power between any transmitter and any receiver.

Note that the central controller must know the active transmitters at any point in time to predict correct interference at all times; this information may be too costly to obtain, but time sampling may be done. Since downlink volume presently dominates WLAN traffic, this paper considers a case where all APs are actively transmitting downlink traffic. It is reasonable to assume that frequency allocation is optimized with respect

[1]Several indoor position-location approaches, based on signal strength sensing, are widely known today and used in some WLANs [15], [16]. Other triangulation methods can also be used to locate a client. Modern cellular handsets are equipped with GPS chips or other position-location technologies. The state-of-the-art GPS can work not only outdoors but also indoors; various vendors, e.g., Metris and SnapTrack, provide indoor GPS solutions. For example, the indoor GPS technology by Metris can be compared with the matrix of satellites that create the GPS; instead of satellites, Metris' indoor GPS uses small infrared laser transmitters that emit laser pulses to create a measurement universe. Based on the timing of the light pulses received by photo detectors, angle and positions may be computed.

to this most active case, since in this case, frequency allocation is most crucial for interference mitigation. Simulations in Section V show that our algorithms also perform well in scenarios with both downlink and uplink traffic and with different levels of AP activity. In this paper, we consider perfect site-specific knowledge; in other words, we assume that the actual path loss between any transmitter and receiver can be correctly predicted by site-specific knowledge. The work in [3] and [4] shows that site-specific models can achieve remarkably good predictions (a zero mean error and a standard deviation of 3–4 dB). The study of the effect of imperfect predictions of channel gains is also an ongoing and future work.

B. Organization

Section II introduces the system model, notation, and assumptions and describes in detail our notion of *weighted interference*. The three proposed measurement-based algorithms, denoted *No-Coord*, *Local-Coord*, and *Global-Coord*, have different mechanisms for iteratively switching frequency channels to reduce the weighted interference seen in a single cell, a group of nearby cells, and all cells, respectively, where a *cell* means an AP (or base station) and its associated users. Section III presents the mechanisms used by the three measurement-based algorithms and their convergence. Section IV presents two site-specific knowledge-based algorithms. Section V shows by simulation that our algorithms substantially outperform [8]–[12].

II. System Model, Notation, and Assumptions

We begin by describing the basic notation; then, the first section describes *weighted interference*, which is a metric used in the three proposed measurement-based algorithms to capture the overall interference of each cell. The second section defines the notation exclusively used for the proposed measurement-based *Local-Coord* algorithm. The third section describes assumptions used in site-specific knowledge-based algorithms.

1) Basic Notation: Suppose M APs, indexed by $\mathbb{M} = \{1, 2, \ldots, M\}$, operate on K orthogonal frequency channels, indexed by $\mathbb{K} = \{1, 2, \ldots, K\}$. We index users (or clients) by $\mathbb{L} = \{1, 2, \ldots, L\}$. We denote the identity of an AP and a client by a_m $(m \in \mathbb{M})$ and c_l $(l \in \mathbb{L})$, respectively. We assume for this paper that the locations of the APs and the clients do not vary with time, and we assume that no APs or users are at the same location, although the algorithms given here also apply for mobile APs and/or clients. Let $\mathbb{L}_m$ $(\mathbb{L}_m \subseteq \mathbb{L})$ denote the set of users that are associated with the AP a_m. We assume every user is associated with a single AP and define *cell* $\mathbb{Z}_m = \{a_m\} \cup \{c_l : l \in \mathbb{L}_m\}$. Let f_m $(f_m \in \mathbb{K})$ denote the channel that a_m operates on, and let $\vec{f} = (f_1, f_2, \ldots, f_M)$ denote the channels of all M APs.

A. Weighted Interference for Measurement-Based Algorithms

In brief, the weighted interference of each cell (e.g., $\mathbb{Z}_m$) is intended to capture the overall interference seen in the cell and is therefore defined as a weighted sum of the average *in situ* measurements at a_m and at all clients associated with a_m, i.e., at every $u \in \mathbb{Z}_m$. We propose that a_m or the clients associated with a_m measure their *in situ* interference power when there is no traffic within cell $\mathbb{Z}_m$, i.e., a_m is neither transmitting nor receiving data. The average *in situ* measured interference power at u (for every $u \in \mathbb{Z}_m$) on channel k is denoted $I_k^u(\vec{f})$. The averaging period is a design choice and could be the same as the period that an AP switches its channel, e.g., 1, 2, or 5 min. $I_k^u(\vec{f})$ is lower bounded by the noise floor. The *weighted interference function* for $\mathbb{Z}_m$ on channel k is defined by

$$W_k^m(\vec{f}) = \sum_{u \in \mathbb{Z}_m} B_k^u\left(I_k^u(\vec{f})\right), \qquad k \in \mathbb{K} \tag{1}$$

where $B_k^u(\cdot)$ is a nonnegative and nondecreasing function that captures the weight of the *in situ* measurement at u. We require that $W_k^m(\vec{f}) > 0$ to capture the existence of the noise floor in the real world. $B_k^u(\cdot)$ should be designed to reflect the difference of clients' traffic demands, signal strengths, and uplink and downlink traffic volume. In practice, clients report the measurements to a_m either periodically or upon request from a_m; then, $W_k^m(\vec{f})$ can be computed at a_m.

In Section III-E, we will show that two of our proposed algorithms (namely, *Local-Coord* and *Global-Coord*) converge if the weighted interference function has the general form in (1). We will later introduce two simplified forms of $W_k^m(\cdot)$ representing practical metrics. The first form, denoted *user based*, places different weights on the *in situ* interference measurements at clients based on the traffic volume and the signal strength at each client. The *user-based* form captures the performance of downlink transmission, which is appropriate for WLANs since traffic measurements show that downlink traffic volume accounts for more than 84% of total (uplink plus downlink) traffic volume [6]. The second form, denoted *AP based*, includes the interference measurements at APs only. The *AP-based* form can be viewed as a simplified version of the *user-based* form by considering that all users have the same traffic volume and signal strength.

1) User Based: The *user-based* weighted interference function for $\mathbb{Z}_m$ is defined by

$$W_k^{(U),m}(\vec{f}) = \sum_{l \in \mathbb{L}_m} \frac{Y_{c_l,a_m}}{S_{c_l,a_m}} \cdot I_k^{c_l}(\vec{f}) \tag{2}$$

where S_{c_l,a_m} denotes the average received signal power[2] from a_m to c_l, and Y_{c_l,a_m} denotes the average traffic volume from a_m to c_l. We incorporate the inverse of S_{c_l,a_m} in (2) because a client with a stronger S_{c_l,a_m} has higher tolerance to interference and, thus, should contribute less to the overall weighted interference. Y_{c_l,a_m} is included in (2) as a scaling factor, since a client with higher traffic volume should be more important for the weighted interference. In practice, some users may be

[2]Note that c_l cannot directly measure S_{c_l,a_m} but can estimate S_{c_l,a_m} as follows: The average *in situ* SINR at c_l can be measured at c_l when a_m is transmitting to c_l and is denoted γ_l. We assume that the interference at c_l is the same whether a_m is transmitting to c_l or a_m is idle, i.e., the interference at c_l is always $I_{f_m}^{c_l}(\vec{f})$. Then, we estimate $S_{c_l,a_m} = \gamma_l \cdot I_{f_m}^{c_l}(\vec{f})$.

sampled to reduce the complexity of computing (2), i.e., the summation in (2) may be over a subset of $\mathbb{L}_m$.

2) AP Based: The *AP-based* weighted interference function for a_m is defined by

$$W_k^{(A),m}(\vec{f}) = I_k^{a_m}(\vec{f}). \tag{3}$$

B. Interfering Cells for the Local-Coord Algorithm

When an AP switches its channel, nearby cells may see substantial changes in their weighted interference. The notation of such cells is presented in the following discussion and will be used in describing the proposed *Local-Coord* algorithm in Section III-B. Cell $\mathbb{Z}_n$ is said to be interfered by cell $\mathbb{Z}_m$ (or $\mathbb{Z}_m$ interferes with $\mathbb{Z}_n$) if and only if a_m or a user associated with a_m induces nonnegligible interference (e.g., the interference power at the receiver is higher than the noise floor) at a_n or a user associated with a_n. We define $\mathbb{G}_m$ as *the set of cells interfered by* $\mathbb{Z}_m$, i.e., $\mathbb{G}_m = \{n : \mathbb{Z}_m$ interferes with $\mathbb{Z}_n$ given that a_m and a_n are on the same channel$\}$. The subset of $\mathbb{G}_m$ that is on channel k is denoted $\mathbb{G}_{m,k}(\vec{f})$. Suppose a_m switches from channel k to k', the cells that see changes in their weighted interference are $\mathbb{Z}_m$ and the cells indexed by $\mathbb{G}_{m,k}(\vec{f})$ and $\mathbb{G}_{m,k'}(\vec{f})$. We define $\mathbb{H}_{m,k,k'}(\vec{f}) \equiv \{m\} \cup \mathbb{G}_{m,k}(\vec{f}) \cup \mathbb{G}_{m,k'}(\vec{f})$; we will see in Section III-B that the weighted interference of the cells indexed by $\mathbb{H}_{m,k,k'}(\vec{f})$ is examined by *Local-Coord* if a_m switches from channel k to k'.

We define $\mathbb{V}_m$ as the set of the indexes of cells that interfere with $\mathbb{Z}_m$ or the cells indexed by $\mathbb{G}_m$, i.e., $i \in \mathbb{V}_m$ if and only if there exists $j \in \{m\} \cup \mathbb{G}_m$ such that $\mathbb{Z}_i$ interferes with $\mathbb{Z}_j$. The notion of $\mathbb{V}_m$ will be used for describing a parallel protocol in Section III-B. Suppose we are given the locations of all controlled APs and possible locations of clients; then, the sets of $\mathbb{G}_m$ and $\mathbb{V}_m$ can be precomputed and preconfigured in the controlled APs or a central network controller that communicates with the controlled APs, using radio propagation prediction models as described in [3]–[6], [14], and [18].

C. Assumptions for Site-Specific Knowledge-Based Algorithms

We assume that the central network controller periodically (e.g., every 5 min) requires the APs to stop transmitting for a short duration of time (e.g., 1 s). In this duration, APs take turns in requiring all users associated with them to perform measurements of background interference, which refers to both the noise floor and rogue interference from RF devices outside the controlled network. Note that each user needs to measure the background interference for all available frequency channels. The users then feed back to APs these measured background interference. Site-specific knowledge, along with measurements of background interference, makes the estimations of SINR at users or APs much more accurate.

We assume *perfect* site-specific knowledge; in other words, we assume that the actual path loss between any transmitter and receiver can be correctly predicted by site-specific knowledge. A study of the effect of *imperfect* predictions of channel gains is an ongoing and future work.

III. Three Measurement-Based Algorithms

The three proposed algorithms all have an iterative nature. At each point in time (predefined, randomly chosen, or determined at runtime), e.g., every 1, 2, or 5 min, one *iteration* of channel switching takes place where one or more APs switch their frequency channels according to mechanisms that are specific to each algorithm, whereas other APs stay on their current channels. The channel switching time in hardware is several milliseconds and is thus negligible as compared to the interval between two iterations. APs and clients measure and average their *in situ* interference between successive iterations. Iterations keep taking place on different AP(s) until the channel allocations converge; of course, when users move or the propagation environment changes, the algorithms will switch channels again until the frequency allocations reach another convergence point. We next describe the different conditions under which each of the algorithms may switch a representative AP a_m's channel from $k = f_m$ to $k' = f'_m$. Throughout this paper, $\vec{f'} \in \mathbb{K}^M$ denotes a vector of channels selected by APs after the representative AP a_m moves from channel f_m to f'_m. Hence, $\vec{f'}$ differs from $\vec{f}$ in only the mth element.

A. No-Coord Algorithm

A representative AP a_m switches from its current channel f_m to f'_m only if the weighted interference on the new channel f'_m is lower, i.e., the following *No-Coord* condition holds:

$$W_{f_m}^m(\vec{f}) > W_{f'_m}^m(\vec{f'}). \tag{4}$$

This algorithm is denoted *No-Coord*, because a_m makes a greedy channel selection without coordination with other APs.

B. Local-Coord Algorithm

If a_m switches from channel k to k', only $\mathbb{Z}_m$ and the cells indexed by $\mathbb{G}_{m,k}(\vec{f})$ and $\mathbb{G}_{m,k'}(\vec{f})$ see changes in their weighted interference. The AP a_m switches from channel k to k' if the max weighted interference seen by these cells decreases after channel switching, i.e., the following *Local-Coord* condition holds:

$$\max_{i \in \mathbb{H}_{m,k,k'}(\vec{f})} W_{f_i}^i(\vec{f}) > \max_{i \in \mathbb{H}_{m,k,k'}(\vec{f})} W_{f'_i}^i(\vec{f'}) \tag{5}$$

where $\mathbb{H}_{m,k,k'}(\vec{f})$ has been defined in Section II-B. This algorithm is denoted *Local-Coord*, since a_m needs to *locally* coordinate with the APs indexed by $\mathbb{G}_{m,k}(\vec{f})$ and $\mathbb{G}_{m,k'}(\vec{f})$ via a wired backbone network for channel switching.

For example, Fig. 1 depicts the cells that see changes in weighted interference before and after AP-1 switches its channel. Since the max weighted interference seen by cells 1–4 decreases, AP-1 can switch to the new channel.

Since coordination among APs is confined in a local area, multiple APs that are far apart enough can simultaneously

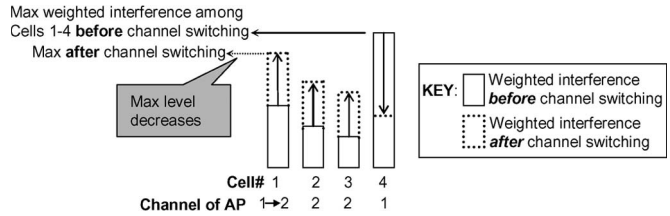

Fig. 1. Decrease of the max weighted interference seen by cells 1–4 *before* and *after* AP-1 switches from channel 1 to 2.

(a) Suppose a timer triggers a_m to consider initiating a channel switching. Then a_m will do the following procedure.

if $\psi_m = 0$ **then**
  Phase 1: Set $\psi_m = -1$ and send requests to lock all APs indexed by $\mathbb{V}_m$, i.e., $\{a_n : n \in \mathbb{V}_m\}$.
  Phase 2: Wait for replies from $\{a_n : n \in \mathbb{V}_m\}$.
  if If the replies indicate that $\{a_n : n \in \mathbb{V}_m\}$ were all successfully locked by a_m **then**
    a_m switches its channel from k to k', and stays at k' if (5) is satisfied; otherwise, a_m switches back to channel k.
    Send messages to unlock $\{a_n : n \in \mathbb{V}_m\}$.
  else
    Send messages to unlock the APs among $\{a_n : n \in \mathbb{V}_m\}$ that were just successfully locked by a_m. (Do not need to unlock the APs that could not be locked by a_m.)
  end if
  Set $\psi_m = 0$.
end if

(b) Upon receiving a *locking request* from a_m, a_n will do the following procedure.

if $\psi_n \neq -1$ **then**
  Increase ψ_n by one.
  Reply to a_m that a_n was successfully locked by a_m.
else
  Reply to a_m that a_n could not be locked.
end if

(c) Upon receiving an *unlocking request* from a_m, a_n will decrease ψ_n by one.

Fig. 2. Protocol for the distributed implementation of *Local-Coord*.

change their channels if a proper inter-AP protocol is employed. In general, the number of APs that can simultaneously change channels grows with the number of total APs; hence, *Local-Coord* is scalable. Fig. 2 presents a distributed protocol implementing *Local-Coord*. We say an AP a_m is *locked* if a_m is not allowed to switch its channel per other APs' requests; if a_m is *unlocked*, a_m may switch its channel. First, we suppose that each AP has an independent random timer that triggers the AP to initiate the process of switching its channel as described in Fig. 2(a). If a_m is locked, a_m will ignore this trigger and wait for the next trigger. The key idea of this protocol is described in Phases 1 and 2 in Fig. 2(a): a_m needs to lock all the APs indexed by $\mathbb{V}_m$ (as defined in Section II-B) before a_m switches to a new channel; then, a_m unlocks those APs after it possibly switches the channels. If any AP indexed by $\mathbb{V}_m$ cannot be locked, a_m cannot switch its channel. The procedure to handle locking and unlocking requests is described in Fig. 2(b) and (c), respectively. An AP can be locked multiple times by different

TABLE I
VARIABLE ψ_m USED IN THE DISTRIBUTED PROTOCOL IN FIG. 2

ψ_m	Channel switching at a_m	Can a_m be locked?
-1	a_m is in the process of switching its channel	No
0	a_m can initiate the process of channel switching	Yes
1 or more	a_m cannot initiate the process of channel switching	Yes

APs; Table I describes ψ_m, which denotes the number of times that a_m was locked. Only when $\psi_m = 0$ can a_m initiate the process of channel switching. When a_m is in the process of switching its channel (denoted by $\psi_m = -1$), it cannot be locked.

Deadlock is a problem that needs to be avoided in such distributed algorithm; in this context, *deadlock* means that two or more APs that have initiated the process of switching their channels are waiting for one other, and thus, none of these APs can ever finish. In the sixth and eighth steps of Fig. 2(a), a_m immediately unlocks other APs whether a_m switches its channel or not; hence, *deadlock never arises in the protocol in* Fig. 2 (see [17] for a detailed description and a proof of deadlock prevention).

C. Global-Coord Algorithm

The AP a_m will switch to a new channel only if the *sum* interference on the new channel is lower (after a_m switches there) than the *sum* interference on its current channel, i.e., the following *Global-Coord* condition holds:

$$\sum_{n:f_n=k} W_k^n(\vec{f}) > \sum_{n:f'_n=k'} W_{k'}^n(\vec{f'}). \tag{6}$$

This algorithm requires global coordination among APs using a central network controller that communicates with all APs and is thus denoted *Global-Coord*.

D. Implementation Concerns

Note that in the descriptions of the three proposed algorithms, some terms in the weighted interference function are unknown before a_m switches to the new channel. An implementation may require a_m to switch to a new channel on a trial basis and then require one or more cells to measure and compute their weighted interference after a_m commits to the switch. Only when all the quantities needed for the channel decisions are known can a_m decide whether switching to the new channel complies with the condition described for each algorithm. If the condition is satisfied, a_m stays on the new channel; otherwise, a_m switches back to the old channel or tries another channel. *No-Coord* requires the weighted interference at cell $\mathbb{Z}_m$; *Local-Coord* requires the weighted interference at cells indexed by $\mathbb{H}_{m,k,k'}(\vec{f})$; and *Global-Coord* requires the weighted interference at all cells.

E. Convergence and Characterization of Convergence Points

Theorem III.1: Consider a particular realization of the locations of APs and users and a weighted interference function of

the form of (1). Given any set of initial AP channel choices, the channel selection process for *Local-Coord* and *Global-Coord* converges in a finite number of steps.

Before characterizing the convergence points for *No-Coord*, *Local-Coord*, and *Global-Coord*, we need some definitions. A vector of frequency allocations denoted by $\vec{f}$ is a *Nash equilibrium* (a concept widely used in game theory [19]) if no single cell can lower its weighted interference by unilaterally changing its channel.

Let $\vec{u} = (u_1, \ldots, u_N)$ and $\vec{u'} = (u'_1, \ldots, u'_N)$ denote the nonincreasing sorted versions of two arbitrary vectors $\vec{v} = (v_1, v_2, \ldots, v_N)$ and $\vec{v'} = (v'_1, v'_2, \ldots, v'_N)$, respectively. We say that $\vec{v}$ lexicographically dominates $\vec{v'}$ (or $\vec{v} \succ \vec{v'}$) if there exists some index j, where $N \geq j \geq 1$ for which $u_j > u'_j$ and $u_i = u'_i$ for all $i < j$. Vectors $\vec{v}$ and $\vec{v'}$ have *the same lexicographic order* if $\vec{u}$ and $\vec{u'}$ are elementwise the same. We say $\vec{v} \succeq \vec{v'}$ if $\vec{v} \succ \vec{v'}$ or $\vec{v}$ and $\vec{v'}$ have the same lexicographic order. We say that a vector of frequency allocations denoted by $\vec{f}$ is a *local lexicographic minimum* with respect to a vector function $\vec{\theta}(\cdot)$, if for any vector of frequency allocations $\vec{f'} \in \mathbb{K}^M$ that differs from $\vec{f}$ in only one element, $\vec{\theta}(\vec{f'}) \succeq \vec{\theta}(\vec{f})$ holds true.

Theorem III.2: Suppose *No-Coord* converges to a frequency allocation $\vec{f}$. Then, $\vec{f}$ is a Nash equilibrium.

Note that *No-Coord* does not always converge, although simulation results show that *No-Coord* converges in most cases. Theorem III.2 is true only for the case where *No-Coord* converges. To resolve the nonconvergence problem of *No-Coord*, one may limit the number of iterations or specify a minimum difference of weighted interference (before and after channel switching) so that *No-Coord* can stop. We next state a technical assumption useful in proving Theorem III.3.

Assumption III.1: Since the weighted interference in (1) takes a continuum of values, it is reasonable to assume that the weighted interference values at different cells or channels are distinct, i.e., $\forall k, j \in \mathbb{K}, \forall m, n \in \mathbb{M}$ such that $k \neq j$ or $m \neq n$, we have $W_k^m(\vec{f}) \neq W_j^n(\vec{f})$ with a probability of 1.

Theorem III.3: Suppose *Local-Coord* or *Global-Coord* converges to a frequency allocation $\vec{f}$. Then, with a probability of 1, $\vec{f}$ is a *local lexicographic minimum* with respect to the vector function $\vec{\alpha}(\cdot)$ as defined in (7) for *Local-Coord*, or with respect to $\vec{\beta}(\cdot)$ as defined in (8) for *Global-Coord*, where

$$\vec{\alpha}(\vec{f}) = \left(W_{f_1}^1(\vec{f}), W_{f_2}^2(\vec{f}), \ldots, W_{f_M}^M(\vec{f})\right) \tag{7}$$

$$\vec{\beta}(\vec{f}) = \left(\sum_{n:f_n=1} W_1^n(\vec{f}), \sum_{n:f_n=2} W_2^n(\vec{f}), \ldots, \sum_{n:f_n=K} W_K^n(\vec{f})\right). \tag{8}$$

IV. Site-Specific Knowledge-Based Algorithms

A. SS-S Formulation

We shall consider optimizing a sum of utility functions for all the users' SINR, assuming all APs are actively transmitting downlink traffic (but not uplink traffic). That is, we optimize the following problem over $\vec{f} \in \mathbb{K}^M$, which is denoted *site-specific SINR (SS-S)* in the rest of this paper:

$$\max_{\vec{f} \in \mathbb{K}^M} \left\{ \sum_{l \in \mathbb{L}} U(\gamma_l) | \right. \tag{9}$$

$$\left. \gamma_l = \frac{S_{c_l, a_m}}{P_{c_l}^i + \sum_{n: f_n = f_m, n \neq m} S_{c_l, a_n}} \qquad \forall l \in \mathbb{L} \right\} \tag{10}$$

where a_m in (10) denotes the AP with which c_l is associated, $P_{c_l}^i$ denotes the background interference power that c_l measures (as described in Section II-C), γ_l denotes the SINR at user c_l [as shown in (10)], and S_{c_l, a_m} denotes the average received signal power from a_m to c_l. Note that the objective in (9) is not optimizing the "sum SINR," since such an objective may favor users that are closer to APs and may cause users that are further away to suffer a low SINR. A fair SINR distribution can be achieved if we optimize the sum of utility functions in (9), where the utility function $U(\cdot)$ in (9) can be any function that is concave, continuously differentiable, and strictly increasing. For example, Mo and Walrand have proposed a class of utility functions that capture different degrees of fairness parameterized by q [20], i.e.,

$$U(\gamma_l) = \begin{cases} (1-q)^{-1} \gamma_l^{(1-q)}, & \text{if } q \neq 1 \\ \log \gamma_l, & \text{if } q = 1 \end{cases}, \gamma_l \in (0, \infty). \tag{11}$$

This family of utility functions is concave, continuously differentiable, and strictly increasing. Intuitively, as q increases, the degree of fairness increases, but the sum SINR decreases. A tradeoff between the sum SINR and fairness of the individual SINR of users that are further away from a serving AP can be observed. By increasing the degree of fairness, we imply that users that are further from APs have a higher SINR (which is needed to provide high throughputs to distant users). The work in [20] shows that if $q \to \infty$, the formulation in (9) becomes a special case that achieves max–min fairness. At max–min fairness, the degree of fairness is the highest; however, the sum SINR is the lowest. Simulation results in Section V show that $q = 2$ may be a good parameter to capture this tradeoff, but this remains a topic for further research. Note that the general form of the weighted interference function in (1) for our measurement-based algorithms could also incorporate utility functions such as those in (11).

As described in Section II, we assume that the APs and/or the users in the controlled network periodically measure the background interference; hence, $P_{c_l}^i$ in (10) is known. We assume that the central network controller has site-specific knowledge and the locations of all APs and users, can predict signal power for any pair of AP and user, and can compute S_{c_l, a_n} for all c_l, a_n in the denominator of (10). Thus, all the quantities in the optimization problem in (9) are known, yet measurement-based algorithms (such as those presented in Section III) do not know each individual component of S_{c_l, a_n} in (10) and, thus, are not able to directly solve (9). Because the optimization in (9) is a

combinatorial problem, there is no fast algorithm (polynomial time) that can solve (9). Therefore, we propose an efficient heuristic in Section IV-C that can find the locally optimal solution of (9); simulations show that the algorithm in Section IV-C outperforms the measurement-based algorithms in Section III, as well as all other frequency allocation algorithms in [8]–[12].

B. SS-R Formulation

The formulation in (9) in Section IV-A strives to provide a fair SINR across users. From the users' perspective, however, the *throughput* may be a better metric than the SINR for users' performance. We now formulate another problem that aims at providing a fair throughput across users, and this formulation may be denoted *site-specific rate (SS-R)*, i.e.,

$$\max_{\vec{f}\in\mathbb{K}^M}\left\{\sum_{l\in\mathbb{L}}U(\chi_l)|\chi_l=\frac{r_l(\gamma_l)}{L_m}\right. \tag{12}$$

$$\left.\gamma_l=\frac{S_{c_l,a_m}}{P^i_{c_l}+\sum_{n:f_n=f_m,n\neq m}S_{c_l,a_n}}\quad\forall l\in\mathbb{L}\right\} \tag{13}$$

where L_m denotes the number of clients that are associated with a_m, χ_l denotes the throughput of c_l from a_m (c_l is associated with a_m), and $r_l(\gamma_l)$ denotes the long-term average data rate that c_l receives from a_m if c_l is the only user associated with a_m; r_l depends on the SINR seen at user c_l, i.e., γ_l, as defined in (10). $r_l(\gamma_l)$ may also be viewed as the achievable capacity between c_l and a_m. We assume that the AP a_m evenly distributes its resource (e.g., time) among its L_m users and therefore has the denominator in (12). There are several ways to model $r_l(\gamma_l)$; for example, we may use Shannon capacity

$$r_l(\gamma_l)=\log_2(1+\gamma_l) \tag{14}$$

or an empirical model, e.g., such as introduced in [6] and [18], to relate the throughput to the SINR, i.e.,

$$r_l(\gamma_l)=T_{\max}\left(1-e^{-A_e(\gamma_l-\gamma_0)}\right) \tag{15}$$

where the three constants $T_{\max}$, A_e, and γ_0 denote the peak throughput, the slope of throughput variation, and the cutoff SINR, respectively, as described in [6]. Note that the model in (15) captures the downlink throughput of a client c_l when all other clients associated with the same AP are idle, and the received SINR of this client c_l is γ_l. In our simulation, we use a time-division multiplexing model for medium access. Hence, at any point of time, an AP is sending data to only one client, and the SINR at this client can be computed by considering interference from all other APs on the same channel. Hence, the model in (15) is valid, as long as we multiply the throughput in (15) by the time fraction that the AP allocates to client c_l. It is known that the IEEE WLAN operates under a carrier sense multiple access/collision avoidance (CSMA/CA) algorithm, and the throughput of a WLAN network is actually more complicated than the simple equation in (15); however, the works in [6] and [18] have shown that the empirical model in (15) can serve as a first-order approximation of WLAN throughputs with high accuracy.

C. Local Optimization Algorithm for SS-S and SS-R

The optimization problems in (9) and (12) are combinatorial; solving them exhaustively requires exponential computation time (exponential in the number of APs). Hence, we present an iterative local optimization procedure that yields rapid and nearly optimal solutions of (9); the same procedure can also solve (12). At the beginning of each iteration, a frequency allocation $\vec{f}$ is given, and at the end of the iteration, we find a better frequency allocation $\vec{g}$ that improves the objective in (9); $\vec{f}$ and $\vec{g}$ may differ in several elements, which means that the channels of several APs may change. During each iteration, we do the following steps: First, we select an AP, e.g., a_m. We find $V-1$ other APs that produce the strongest interference on a_m, assuming these $V-1$ other APs and a_m are on the same channel; for example, $V=7$ implies that we find six other APs that are in the vicinity of a_m so that they will likely interfere with a_m's clients the most. We try all possible K^V permutations of channels for these V APs while fixing the channels at the other $M-V$ APs. We can find the best out of the K^V permutations so that (9) is maximized and is strictly larger than the value before this iteration; then, we change the corresponding V elements in $\vec{f}$ and thus form $\vec{g}$. If these V APs have operated on the best channel allocation before this iteration, we have $\vec{f}=\vec{g}$; in this case, another AP (instead of a_m) and its $V-1$ neighboring APs will be selected to restart this iteration. This iterative algorithm runs until every set of V neighboring APs reaches the best frequency allocation. This iterative algorithm converges in a finite number of steps, since the number of channel permutations is finite, and each iteration strictly increases the objective in (9). In practice, one may limit the number of iterations or specify a minimum difference of weighted interference so that the iterations can be finished in a reasonable amount of time (e.g., 1, 5, or 30 s). We expect that the channel allocation found by this local optimization algorithm will be close to the optimum if V is large enough, since the exhaustive search can explore more possible allocations with a larger V. Nevertheless, the simulations in Section V show that this local optimization algorithm with $V=7$ outperforms all other algorithms [8]–[12].

The previously proposed algorithm solves the *SS-S* formulation in (9) and the *SS-R* formulation in (12). When it is applied to solve *SS-S*, we refer to the algorithm as the *SS-S* algorithm; similarly, when the algorithm is used to maximize the throughput (rate), we refer to it as the *SS-R* algorithm.

V. Simulation Setup and Results

We shall begin by describing our simulation setup in Section V-A. Then, in Section V-B, we present and discuss our simulation results.

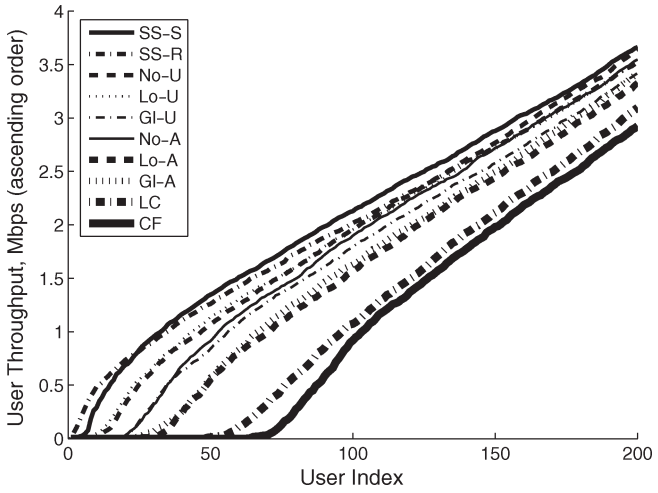

Fig. 3. *User throughput* (in megabits per second) comparison in a setting with APs on a *uniform* 10-by-10 layout, 400 users, and ten rogue RF interferers. Only the 200 users with lower throughputs are shown.

A. Simulation Setup

No-Coord, *Local-Coord*, and *Global-Coord*, along with the *user-based* weighted interference function in (2) and the *AP-based* weighted interference function in (3), yield six algorithms, namely, *No-U*, *Lo-U*, *Gl-U*, *No-A*, *Lo-A*, and *Gl-A*. The algorithm in [11], denoted *CF*, has been shown to outperform [8]–[10]. Hence, we compare our proposed algorithms (the preceding six combinations, as well as *SS-S* and *SS-R*) against *CF* and the algorithm in [12], which is denoted *LC*. We set the number of orthogonal channels K to 3 to represent 802.11b/g; other larger values of K produce very similar trends as to those shown in Figs. 3–5, making our approach applicable to cellular networks and 802.11a. We assume that each AP can source a maximum of 54 Mb/s per the 802.11g standard. We consider three network sizes, three levels of rogue interference, and two network topologies, and thus have 18 combinations $(3\times3\times2)$, as shown in the x-axis of Fig. 4. For each of the 18 combinations, we randomly generate ten independent cases and compute the average. The three network sizes include a 4-by-4 AP layout with 64 users, a 7-by-7 layout with 196 users, and a 10-by-10 layout with 400 users. Each AP may be associated with a different number of users; the average number of users for each AP is four for all three network sizes. We consider low, medium, and high interference from rogue interferers, where the ratios of the number of rogue interferers to the number of APs are 10%, 40%, and 70%, respectively. We consider a uniform topology, where APs are regularly located on corners of hexagons with less than 5 m of random perturbation, as illustrated in Fig. 6(a), and a nonuniform topology, where APs are perturbed from the uniform layout by a random distance (up to 25% of separation), as shown in Fig. 6(b). The separation between adjacent APs is 240 m. The path loss exponent is set to be 3. We set a constant transmit power of 10 mW for every AP and client. The noise floor is set to be 10 dB above the thermal noise to properly represent the RF environment [21]; the thermal noise is modeled as kT_0B, where k is the Boltzmann's constant ($k = 1.38 \times 10^{-23}$ J/K), T_0 is the ambient room temperature (typically taken as 300 K), and B is the equivalent bandwidth of the measuring device ($B = 30$ MHz for the bandwidth of IEEE 802.11b/g systems). We set the fairness parameter q as 2.

B. Simulation Results and Discussions for Two Kinds of Traffic Scenarios

We consider results for two traffic scenarios. In the first scenario (Section V-B1), we present results for saturated downlink networks, i.e., all APs are transmitting downlink traffic. Then, in the second scenario (Section V-B2), we present results for networks with both uplink and downlink traffic.

1) Downlink Only: Fig. 3 shows the users' throughputs (in ascending order) resulting from different algorithms for 100 controlled APs and 400 users with ten rogues. Note that the algorithms yield very different throughputs for the first 200 users with lower throughputs but less different for the last 200 users; hence, we emphasize the first 200 users in Fig. 3; more detailed results can be found in [17]. Other scenarios with different numbers of APs, users, and rogue interferers yield similar throughput trends, as in Fig. 3, and are therefore omitted for the sake of brevity. Our proposed algorithms outperform the best algorithms in the literature, i.e., *LC* and *CF*; particularly, the site-specific knowledge-based algorithms have the best performance. Fig. 3 shows that *SS-S* and *SS-R* outperform *LC* by 16.8% and 13.1% in terms of the mean user throughput, 18.5% and 13.6% in terms of median, 97.6% and 87.1% in terms of 25th percentile, 204% and 188% in terms of 20th percentile, and 1180% and 1110% in terms of 15th percentile user throughputs. Although *Lo-U* achieves lower throughputs than *SS-S* or *SS-R*, yet *Lo-U* is the best measurement-based algorithm, particularly good at uplifting throughputs for users with low throughputs. Fig. 3 shows that *Lo-U* outperforms *LC* by 12.9%, 14.3%, 81.4%, 168%, and 1010% in terms of mean, 50th percentile, 25th percentile, 20th percentile, and 15th percentile user throughputs, respectively.

In Fig. 3, *SS-R* yields the highest 5th and 3rd percentile throughputs. Generally, *SS-R* sacrifices the users with higher throughputs to improve the users with lower throughputs. Although *SS-R* is worse than *SS-S* for users with high throughputs, *SS-R* is still better than *Lo-U*, which is the best measurement-based algorithm. Our algorithms yield enormous throughput gains, particularly for users with low throughputs. Fig. 4 shows the percentage of users with throughputs more than 512 kb/s in various scenarios. In Fig. 4, our algorithms enable more users to operate above 512 kb/s irrespective of the number of APs and rogues; this trend is also true for other throughput thresholds (other than 512 kb/s). Fig. 4 shows that *SS-R* and *SS-S* accommodate up to 18.8% more users than *LC* or *CF*.

2) Both Downlink and Uplink: As described in the preceding discussion, we assumed all traffic was *downlink* and optimized the frequency allocation for the *most active case*, where *all APs are transmitting downlink traffic*. It is reasonable to optimize frequency allocation for this most active case, since in this case, frequency allocation is most crucial for interference mitigation at each user. In this section, we examine

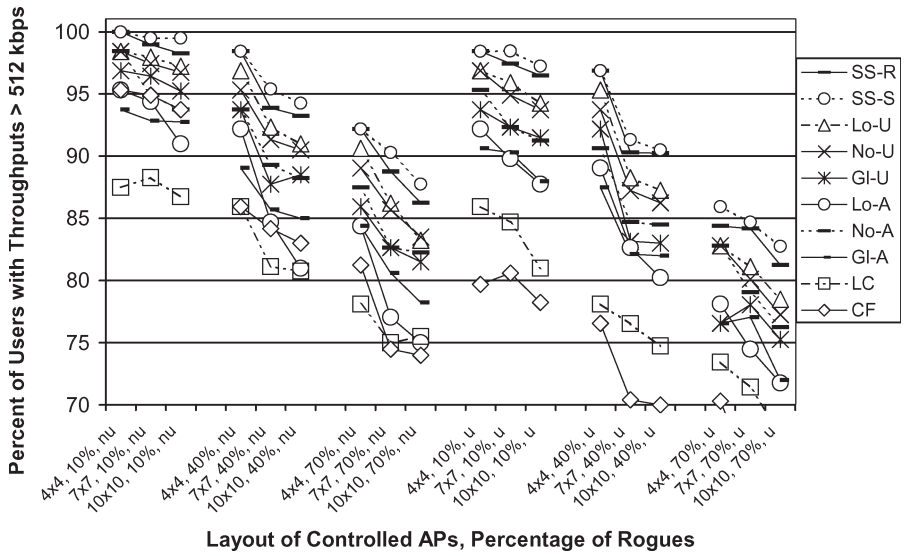

Fig. 4. Percent of users that have throughputs higher than 512 kb/s. The x-axis represents the layout of controlled APs and the percentage of rogue APs compared with the controlled APs. *Nonuniform* and *uniform* AP layouts are denoted "nu" and "u," respectively.

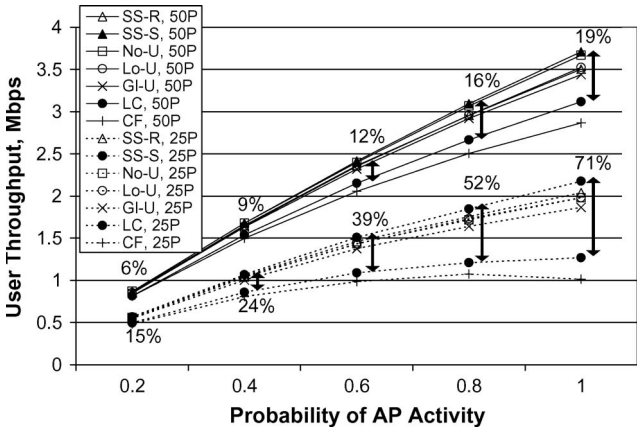

Fig. 5. Fiftieth and 25th percentile user throughputs (50P and 25P), respectively, including both downlink and uplink traffic, for 400 users on a 10-by-10 uniform AP layout with ten rogues. Vertical arrows and numbers beside the arrows depict the gains of *SS-S* over *LC* in percent (some arrows are omitted when the associated gains are relatively small).

the performance of the optimized frequency allocations in the presence of both downlink and uplink traffic. It has been shown in [22] that uplink and downlink capacities in multiple cells are mutually coupled due to intercell interference, and no system-level analytic model has been found to model activities of multiple APs. In our next simulation, we consider that time is slotted and propose an approximate probabilistic model where APs independently choose one of the three possible activity states at each time slot. An AP can be transmitting downlink traffic, receiving uplink traffic, or idle, with probabilities p_d, p_u, and $p_i = 1 - p_d - p_u$, respectively. For any AP that is transmitting downlink traffic or receiving uplink traffic at a certain time slot, a user is randomly chosen (with a uniform probability distribution) out of all the users associated with this AP to be the recipient or the sender of the traffic. We fix the ratio of p_d to p_u as 5 : 1 [6] and simulate five cases where $p_d + p_u$ (the probability that an AP is active) is 0.2, 0.4, . . ., 1.0, respectively. We intend to see the effect of $p_d + p_u$ on the performance of the proposed algorithms. The assumption that the activity of each AP is independent from the other APs simplifies the simulations and provides a rule of thumb for the performance comparison. Fig. 5 shows that our algorithms consistently yield throughput gains (including both downlink and uplink) irrespective of the probability of AP activity; particularly, the gains are high (up to 71% for the 25th percentile throughput and 19% for the median throughput) when APs are highly active (i.e., when the network traffic load is heavy). In Fig. 5, we still see the same trend as in Fig. 3 that *SS-S* and *SS-R* have the best performance in providing high throughputs for users who suffer the lowest throughputs.

VI. Conclusion

A central network controller with site-specific knowledge can predict the path loss between any AP and client and therefore predict the impact of the SINR and throughput on every AP and user when the channel of any AP is changed. This site-specific knowledge leads to vast network improvements, which we have demonstrated by using two site-specific algorithms that can incorporate the importance of fairness across users. Our proposed algorithms are particularly useful when the traffic load of the network is high and APs are highly active. The two algorithms, namely, *SS-S* and *SS-R*, are better in uplifting the throughputs of users that suffer low throughputs when particular utility functions are chosen. Judicious selection of utility function is a topic of future research. We believe that site-specific knowledge is also useful for other wireless communication problems in both cellular networks and WLANs, which will be validated by ongoing and future work; for example, the work in [23] uses site-specific knowledge to perform load balancing in wireless networks.

When site-specific knowledge is not available, the proposed measurement-based algorithms are good alternatives. Among

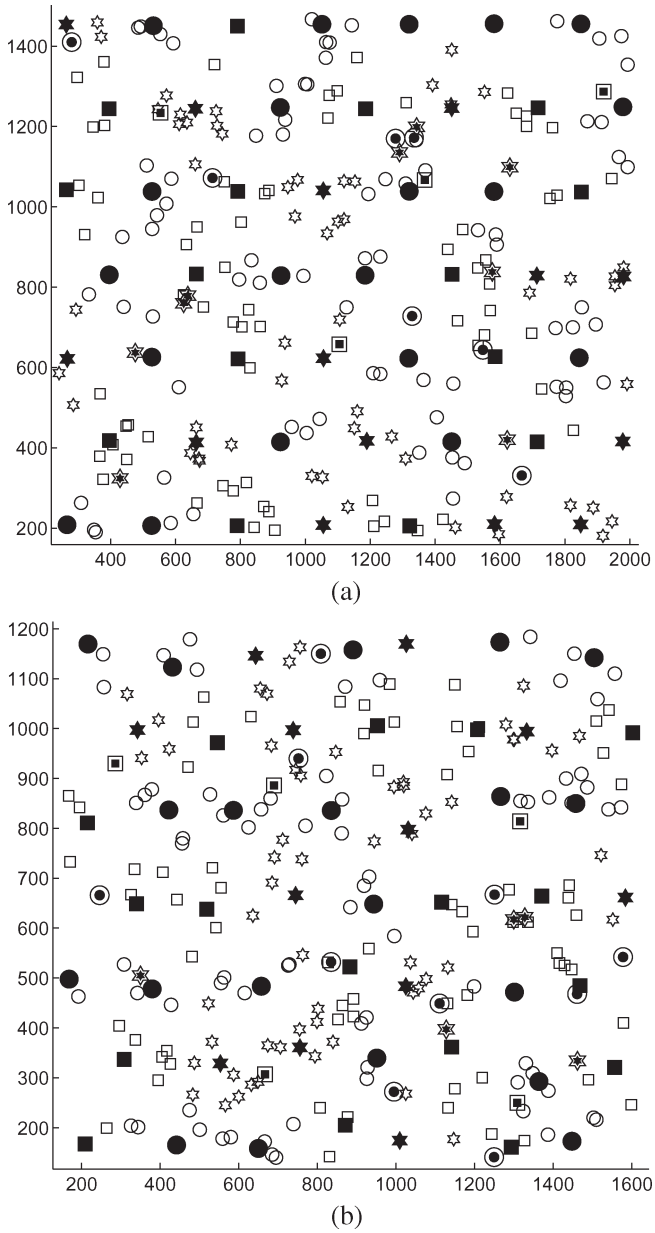

Fig. 6. Frequency allocation examples for 49 APs on a 7-by-7 *nonuniform* or *uniform* topology. Three kinds of objects (squares, stars, and circles) signify three orthogonal frequency channels. Filled back objects denote 49 APs, hollow objects denote 196 users; and double-layered objects with inner part filled with black denote 20 rogues. The units of the x- and y-axes are meters.

the three measurement-based algorithms, *Local-Coord* is the best in uplifting the throughputs of users that suffer low throughputs. For *Local-Coord*, a scalable distributed protocol is given, and the convergence is proven; hence, *Local-Coord* is our best measurement-based algorithm for frequency allocation in wireless networks. If coordination among APs cannot be realized as required in *Local-Coord*, *No-Coord* is also a good option, since it does not require coordination among APs. Although *No-Coord* is not guaranteed to converge, simulations show that it converges in most cases and has a comparable throughput gain as *Local-Coord*. We present practical approaches to implement these measurement-based algorithms.

APPENDIX
PROOFS OF THEOREMS III.1–III.3

Lemma A.1

Suppose two vectors $\vec{v} = (v_1, v_2, \ldots, v_N)$ and $\vec{v'} = (v'_1, v'_2, \ldots, v'_N)$ differ in at least one element. Assume all elements in $\vec{v}$ are distinct and so are those in $\vec{v'}$. Let $\mathbb{D}$ denote indexes where $\vec{v}$ and $\vec{v'}$ differ, i.e., $\mathbb{D} = \{i : v_i \neq v'_i\}$. Then, we have $\vec{v} \succ \vec{v'}$ if $\max_{i\in\mathbb{D}} v_i > \max_{i\in\mathbb{D}} v'_i$.

Proof sketch: We sort the elements of $\vec{v}$ and $\vec{v'}$, respectively, in descending order and compare their elements one by one from the largest to the smallest. Then, the first different pair of elements between the two sorted vectors is $\max_{i\in\mathbb{D}} v_i$ and $\max_{i\in\mathbb{D}} v'_i$, respectively. Since $\max_{i\in\mathbb{D}} v_i > \max_{i\in\mathbb{D}} v'_i$, we have $\vec{v} \succ \vec{v'}$ according to the definition of lexicographic order in Section III-E. A detailed proof is given in [17]. ■

Lemma A.2

Suppose a_m is a representative AP switching its channel from k to k' according to the *Local-Coord* condition in (5) or *Global-Coord* condition in (6), and the channels of all the other APs remain unchanged. Then, we have $\vec{\alpha}(\vec{f}) \succ \vec{\alpha}(\vec{f'})$ for *Local-Coord* or $\vec{\beta}(\vec{f}) \succ \vec{\beta}(\vec{f'})$ for *Global-Coord* [$\vec{\alpha}(\vec{f})$ defined in (7) and $\vec{\beta}(\vec{f})$ defined in (8)].

Proof: If a_m switches from channel k to k', only the cells indexed by $\mathbb{H}_{m,k,k'}(\vec{f})$ see changes in their weighted interference. Note that the nth element of $\vec{\alpha}(\vec{f})$ signifies the weighted interference of $\mathbb{Z}_n$. Hence, the different elements between $\vec{\alpha}(\vec{f})$ and $\vec{\alpha}(\vec{f'})$ are those indexed by $\mathbb{H}_{m,k,k'}(\vec{f})$. According to Lemma A.1, it suffices to show that the maximum of these different elements in $\vec{\alpha}(\vec{f})$ is greater than the maximum of those in $\vec{\alpha}(\vec{f'})$, i.e., $\max_{i\in\mathbb{H}_{m,k,k'}(\vec{f})} W^i_{f_i}(\vec{f}) > \max_{i\in\mathbb{H}_{m,k,k'}(\vec{f})} W^i_{f'_i}(\vec{f'})$, which is equal to the *Local-Coord* condition in (5). Hence, the proof for *Local-Coord* is done. The proof for *Global-Coord* is similar and is omitted for the sake of brevity (see [17] for the proof). ■

Proof of Theorem III.1: We will first prove the convergence of *Local-Coord*. We form a directed graph $\mathcal{G}$ with all possible channel vectors $\vec{f}$ as *nodes* (hence the number of nodes is finite) and all channel adjustments that satisfy the *Local-Coord* condition in (5) as *edges*, assuming only one AP switches its channel at any point of time. We will show that this graph is *acyclic*; then, since $\mathcal{G}$ is acyclic and finite, any initial node will converge to a sink in a finite number of steps of channel adjustments. Note that the lexicographic order possesses the transitive property, that is, if $\vec{v} \succ \vec{v'}$ and $\vec{v'} \succ \vec{v''}$, then $\vec{v} \succ \vec{v''}$ [24]. Suppose *there exists a cycle on* $\mathcal{G}$, and $\vec{f^0}, \vec{f^1}, \vec{f^2}, \ldots$ are nodes on this cycle. As we travel through this cycle once, we will see that $\vec{\alpha}(\vec{f^0}) \succ \vec{\alpha}(\vec{f^1}) \succ \vec{\alpha}(\vec{f^2}) \succ \cdots \succ \vec{\alpha}(\vec{f^0})$ according to Lemma A.2. This implies $\vec{\alpha}(\vec{f^0}) \succ \vec{\alpha}(\vec{f^0})$ according to the transitive property, which is a contradiction since $\vec{\alpha}(\vec{f^0})$ does *not* lexicographically dominate itself. Therefore, $\mathcal{G}$ is acyclic, and the proof is done. The proof of *Global-Coord* is the same as the preceding proof except that the edges of $\mathcal{G}$ are

the channel adjustments satisfying the *Global-Coord* condition in (6), and $\vec{\alpha}(\cdot)$ is replaced with $\vec{\beta}(\cdot)$. ■

Proof of Theorem III.2: Suppose *No-Coord* converges at a frequency allocation $\vec{f}$, but $\vec{f}$ is *not* a Nash equilibrium. Then, there exists at least one AP, e.g., a_m, and one channel f'_m ($f'_m \neq f_m$) so that a_m can switch from its current channel f_m to f'_m to strictly decrease the weighted interference of $\mathbb{Z}_m$. Then, the frequency allocation has not converged, since a_m can switch to channel f'_m according to the *No-Coord* condition in (4). This proof is done by contradiction. ■

Proof of Theorem III.3: Recall from the Proof of Lemma A.2 that $\vec{\alpha}(\vec{f})$ differs from $\vec{\alpha}(\vec{f}')$ only in the elements indexed by $\mathbb{H}_{m,k,k'}(\vec{f})$. To prove that $\vec{\alpha}(\vec{f}') \succeq \vec{\alpha}(\vec{f})$ holds with a probability of 1, it suffices to show that

$$\max_{i \in \mathbb{H}_{m,k,k'}(\vec{f})} W^i_{f'_i}(\vec{f}') > \max_{i \in \mathbb{H}_{m,k,k'}(\vec{f})} W^i_{f_i}(\vec{f}) \qquad (16)$$

holds with a probability of 1, according to Lemma A.1. Since *Local-Coord* converges at $\vec{f}$, no AP can move to a new channel so that the *Local-Coord* condition in (5) is satisfied. Hence, for every AP a_m (e.g., it is currently on channel k) and every new channel k' ($k' \neq k$), the converse of (5) holds. The inequality in the converse of (5) holds with a probability of 1 according to Assumption III.1 and is the same as (16); thus, the proof is done. The proof for *Global-Coord* is similar and is omitted for the sake of brevity (see [17] for the proof). ■

REFERENCES

[1] J. K. Chen, T. S. Rappaport, and G. de Veciana, "Site specific knowledge for improving frequency allocations in wireless LAN and cellular networks," in *Proc. IEEE Veh. Technol. Conf.*, Baltimore, MD, Oct. 2007, pp. 1431–1435.

[2] J. K. Chen, G. de Veciana, and T. S. Rappaport, "Improved measurement-based frequency allocation algorithms for wireless networks," in *Proc. IEEE GLOBECOM*, Washington, DC, Nov. 2007, pp. 4790–4795.

[3] S. Y. Seidel and T. S. Rappaport, "914 MHz path loss prediction models for indoor wireless communications in multifloored buildings," *IEEE Trans. Antennas Propag.*, vol. 40, no. 2, pp. 207–217, Feb. 1992.

[4] R. R. Skidmore, T. S. Rappaport, and A. L. Abbott, "Interactive coverage region and system design simulation for wireless communication systems in multifloored indoor environments: SMT Plus," in *Proc. IEEE Int. Conf. Universal Pers. Commun.*, Oct. 1996, vol. 2, pp. 646–650.

[5] M. Hassan-Ali and K. Pahlavan, "A new statistical model for site-specific indoor radio propagation prediction based on geometric optics and geometric probability," *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 112–124, Jan. 2002.

[6] C. Na, J. K. Chen, and T. S. Rappaport, "Measured traffic statistics and throughput of IEEE 802.11b public WLAN hotspots with three different applications," *IEEE Trans. Wireless Commun.*, vol. 5, no. 11, pp. 3296–3305, Nov. 2006.

[7] R. R. Skidmore, A. Verstak, N. Ramakrishnan, T. S. Rappaport, L. T. Watson, J. He, S. Varadarajan, C. A. Shaffer, J. Chen, K. K. Bae, J. Jiang, and W. H. Tranter, "Towards integrated PSEs for wireless communications: Experiences with the S^{4}W and SitePlanner projects," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 8, no. 2, pp. 20–34, Apr. 2004.

[8] K. K. Leung and B.-J. Kim, "Frequency assignment for IEEE 802.11 wireless networks," in *Proc. IEEE Veh. Technol. Conf.*, Oct. 2003, vol. 3, pp. 1422–1426.

[9] Y. Lee, K. Kim, and Y. Choi, "Optimization of AP placement and channel assignment in wireless LANs," in *Proc. IEEE Conf. LCN*, Nov. 2002, pp. 831–836.

[10] A. Mishra, S. Banerjee, and W. Arbaugh, "Weighted coloring based channel assignment for WLANs," *ACM Mobile Comput. Commun. Rev.*, vol. 9, no. 3, pp. 19–31, Jul. 2005.

[11] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh, "A client-driven approach for channel management in wireless LANs," in *Proc. IEEE INFOCOM*, Apr. 2006, pp. 1–12.

[12] D. J. Leith and P. Clifford, "A self-managed distributed channel selection algorithm for WLANs," in *Proc. Int. Symp. Model. Optimization Mobile, Ad Hoc Wireless Netw.*, Apr. 2006, pp. 1–9.

[13] C. Na, J. K. Chen, and T. S. Rappaport, "Hotspot traffic statistics and throughput models for several applications," in *Proc. IEEE GLOBECOM*, Nov. 29–Dec. 3 2004, vol. 5, pp. 3257–3263.

[14] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Englewood Cliffs, NJ: Prentice–Hall, 2002.

[15] R. R. Skidmore and T. S. Rappaport, "System and method for measuring and monitoring wireless network performance in campus and indoor environments," U.S. Patent no. 7 096 160, Aug. 22, 2006.

[16] P. Bahl and V. N. Padmanabhan, "A software system for locating mobile users: Design, evaluation, and lessons," Microsoft Res., Redmond, WA, Apr. 2000. Tech. Rep.

[17] J. K. Chen, "Frequency allocation, transmit power control, and load balancing with site specific knowledge for optimizing wireless network performance," Ph.D. dissertation, Univ. Texas, Austin, TX, May 2007.

[18] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Commun. Mag.*, vol. 41, no. 10, pp. 74–80, Oct. 2003.

[19] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA: MIT Press, 1991.

[20] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[21] E. N. Skomal and A. A. Smith, Jr., *Measuring the Radio Frequency Environment*. New York: Van Nostrand, 1985.

[22] T. Bonald, S. Borst, N. Hegde, and A. Proutiére, "Wireless data performance in multi-cell scenarios," in *Proc. ACM SIGMETRICS*, Jun. 2004, pp. 378–387.

[23] J. K. Chen, T. S. Rappaport, and G. de Veciana, "Iterative water-filling for load-balancing in wireless LAN or microcellular networks," in *Proc. IEEE Veh. Technol. Conf.*, May 2006, vol. 1, pp. 117–121.

[24] *Lexicographical order*, Dec. 2008. [Online]. Available: http://en.wikipedia.org/wiki/Lexicographical order

Jeremy K. Chen (S'03–M'07) received the B.S.E.E. degree from the National Taiwan University, Taipei, Taiwan, in 2001 and the M.S. and Ph.D. degrees in electrical and computer engineering from The University of Texas at Austin (UT Austin) in 2004 and 2007, respectively.

He is currently with Google Inc., New York, NY. He was with Qualcomm Flarion Technologies from 2007 to 2008. He was a Research Assistant with the Wireless Networking and Communications Group (WNCG), UT Austin, during 2003–2007. He held summer internships with Wireless Valley Communications (now Motorola) in 2004 and Intumit Technology in 2000. During 1998 and 2001, he was an Undergraduate Research Assistant with Academia Sinica, Taipei, where he codeveloped a cross-platform Chinese-input software package called *Chewing*. He has extensive experience in networks, wireless communications, algorithms, and software engineering.

Dr. Chen was the recipient of the following awards: the Silver Medal in the International Olympics in Informatics in 1997; First Place at the Annual Joint College Entrance Exam in the Republic of China (Taiwan) in 1997; the Asian Championship and World Final's 10th Place (out of 1400) in the ACM International Collegiate Programming Contest in 1999; and the Taiwan Free Software Community Awards in 2003 for the Chinese-input software package *Chewing*, which is now widely used in the Chinese community.

Gustavo de Veciana (S'88–M'94–SM'01–F'09) received the B.S., M.S., and Ph.D. degrees from the University of California, Berkeley, in 1987, 1990, and 1993, respectively, all in electrical engineering.

He is currently a Professor with the Department of Electrical and Computer Engineering, The University of Texas at Austin, where he served as the Associate Director and then the Director of the Wireless Networking and Communications Group (WNCG) during 2004–2008. His research focuses on the design, analysis, and control of telecommunication networks. His current interests include measurement, modeling, and performance evaluation; wireless and sensor networks; and architectures and algorithms to design reliable computing and network systems.

Dr. de Veciana has served as the Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING and as a Cochair of the International Conference on Emerging Networking Experiments and Technologies 2008 (ACM CoNEXT 2008). He was a recipient of the General Motors Foundation Centennial Fellowship in Electrical Engineering and the National Science Foundation CAREER Award in 1996. He was a corecipient of the IEEE William McCalla Best ICCAD Paper Award in 2000 and the Best Paper Award in the *ACM Transactions on Design Automation of Electronic Systems* in 2002–2004.

Theodore S. Rappaport (S'83–M'84–SM'91–F'98) received the B.S., M.S., and Ph.D. degrees from Purdue University, West Lafayette, IN, in 1982, 1984, and 1987, respectively, all in electrical engineering.

He is the William and Bettye Nowlin Chair in Engineering with The University of Texas at Austin (UT Austin) and is the Founding Director of the Wireless Networking and Communications Group (WNCG), UT Austin (http://www.wncg.org/), which is a center he founded in 2002. Prior to joining UT Austin, he was with the Faculty of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, where he founded one of the world's first university research and teaching centers dedicated to the wireless communications field. He has been a pioneer in the fields of radio wave propagation and wireless communication system design, and his work has influenced many international wireless standard bodies. In 1989, he founded TSR Technologies, Inc., which is a cellular radio/PCS software radio manufacturer that he sold in 1993 to what is now CommScope, Inc. In 1995, he founded Wireless Valley Communications, Inc., which is a site-specific wireless network design and management firm that he sold in 2005 to Motorola, Inc. He has testified before the U.S. Congress, has served as an international consultant for the ITU, has consulted for more than 30 major telecommunications firms, and works on many national committees pertaining to communications research and technology policy. He is a highly sought-after consultant and technical expert. He is one of the world's most highly cited authors in the wireless field, having authored or coauthored more than 200 technical papers and several books. He has 100 U.S. and international patents that are either issued or pending.

Prof. Rappaport was elected to serve on the Board of Governors of the IEEE Communications Society (ComSoc) in 2006 and the Board of Governors of the IEEE Vehicular Technology Society (VTS) in 2008. He was the recipient of the IEEE Communications Society Stephen O. Rice Prize Paper Award in 1999 for his pioneering work on site-specific radio-frequency propagation and system design.