MiFi: A Framework for Fairness and QoS Assurance for Current IEEE 802.11 Networks With Multiple Access Points

Yigal Bejerano and Randeep S. Bhatia

***Abstract*—In this paper, we present a framework for providing fair service and supporting quality of service (QoS) requirements in IEEE 802.11 networks with multiple access points (APs). These issues becomes critical as IEEE 802.11 wireless LAN are widely deployed in nationwide networks, linking tens of thousands of "hot-spots" for providing both real-time (voice) and non real-time (data) services to a large population of mobile users. However, both fairness and QoS guarantees cannot be supported in the current 802.11 standard.**

Our system, termed MiFi, relies on centralized coordination of the APs. During any given time of the "contention-free" period only a set of non-interfering APs is activated while the others are silenced. Moreover, the amount of service granted to an AP is proportional to its load and the system's performance is optimized by employing efficient scheduling algorithms. We show that such a system can be implemented without requiring any modification of the underlying MAC protocol standard or the behavior of the mobile stations. Our scheme is complementary to the emerging 802.11e standard for QoS and guarantees to overcome the hidden node and the overlapping cell problems. Our simulations establish that the system supports fairness and hence can provide QoS guarantees for real-time traffic, while maintaining a relative high throughput.

***Index Terms*—Approximation algorithms, fairness, IEEE 802.11, quality of service (QoS), wireless LAN.**

I. Introduction

In RECENT years, we have been witnessing a tremendous gain in popularity of IEEE 802.11 wireless LANs (WLANs), known as WiFi. WLANs are being rapidly deployed all over the world, in cities, enterprises, and university campuses, and they aim to provide both data and real-time services (such as voice and video) to their users. However, to make these efforts a success, many serious shortcomings of WiFi must be overcome [2]. Some of these include the lack of fairness of the WiFi technology and its inability to provide any kind of quality of service guarantees, making it incapable of supporting real-time (RT) services such as voice and video.

Generally speaking, *fairness* is the ability of a network to provide the same level of service to all its users, while *quality of service* (QoS) is the ability of providing a service with some level of assurance for data delivery. This assurance is usually given in terms of guaranteed bandwidth, delay bounds and jitter, which are essential for RT applications like voice. Providing QoS assurance and fairness are two related problems and a system that cannot provide a certain degree of fair service to its users, i.e., minimal allocated bandwidth, cannot provide QoS guarantees. However, both fairness and QoS guarantees cannot be supported in the current 802.11 standard.

The IEEE 802.11 MAC standard defines two operations modes. The first is a "best effort" distributed coordination function (DCF) in which users compete for their opportunity to transmit data during a contention period. This mode is known to exhibit both short and long term unfairness [3], [4]. That is the MAC layer in DCF mode may fail to provide equitable allocation of channel resources to competing stations, which has detrimental effect on RT applications [3]. Numerous schemes have been proposed in the literature to overcome these shortcomings of the DCF mode (see survey papers [5], [6]). Most of these schemes propose one or a combination of the following techniques; modifying the DCF backoff mechanism [7], [4], employing adaptive size for contention window [8], using different interframe spacing (IFS) [9] or various frame sizes [10]. Other papers [11], [12] propose distributed schemes for obtaining "soft" QoS guarantees. Generally, these papers have shown that distributed schemes are capable of providing fair service and supporting RT traffic when the network load is low and when all the stations are in the transmission range of each other. However, these schemes can provide only "soft" QoS assurance without strict bounds on the message delay and jitter [11]. In addition, their efficiency is reduced at heavy load and when high data rate is used [13].

The second operation mode is the point coordination function (PCF) that was designed to support RT traffic. Here, a network access point (AP) periodically initiates contention free periods (CFPs) in which it polls its associated stations. Unlike the DCF mode, the PCF is a centralized access mode and several recent papers [13]–[15] have shown that it is suitable for supporting RT traffic. The "connection-oriented" behavior of the PCF mode allows the network to provide bandwidth and delay guarantees that are necessary to support RT applications. These papers consider only networks with a single access point and assume that all the users are in the transmission range of the access point. However, this property is not sustained in networks with multiple access points that have overlapping transmission ranges or *cells*. In such networks transmission collisions may occur during a CFP as a result of either "hidden nodes" that have not received the beacon messages announcing the beginning of a CFP, the so-called *hidden node* problem, or when two adjacent access points schedule their CFPs simultaneously, known

Manuscript received Janaury 12, 2004; revised April 13, 2005; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor N. Shroff. An extended abstract of this paper appeared in the Proceedings of the IEEE INFOCOM 2004.

The authors are with Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974 USA.

Digital Object Identifier 10.1109/TNET.2006.880161

as the *overlapping cell* problem [16]. Thus, a station may fail to send or receive data when polled during the CFP as a result of interferences from other cells. Recently, some work has addressed the hidden node and the overlapping cell problems by using distributed time synchronization algorithms [16] and by utilizing methods from game theory [17]. However, these schemes cannot ensure either fair service or QoS guarantees. Currently, the IEEE 802.11 committee is finishing a new proposal for adding QoS assurance capabilities to the existing standard, the so called IEEE 802.11-e proposal [18]. Yet, even this proposal does not provide adequate answers for the hidden node and the overlapping cell problems.

In this paper, we present the Managed WiFi system, called *MiFi*, for supporting fairness and QoS in the existing IEEE 802.11 MAC layer. To the best of our knowledge, this is the first comprehensive system that overcomes both the hidden node and the overlapping cell problems in multiple-AP WLAN networks. Our scheme ensures a fair service for the mobile users and provides the required QoS guarantees for RT applications. It is inspired by the recent results of [13]–[15] that have shown that the PCF mode can efficiently be used for supporting RT sessions and for providing fairness in WLAN when all the users are in the transmission range of a *single* AP. In this work, we extend their result also to networks with *multiple* APs. Our system uses a combination of MAC layer techniques and centralized management and coordination of the access points (APs). All the access points simultaneously toggle between a *contention free period* (CFP) using PCF mode and a *contention period* (CP) using DCF mode. During the CFP only nodes that are polled are allowed to transmit, for preventing the hidden node problem. In the CFP, time is divided into equal sized slots such that within each slot only a subset of APs are activated, while the rest are silenced. By ensuring that the APs activated in any slot of the CFP are "non-interfering," our scheme guarantees that the system is free from the overlapping cell and the hidden node problems. Thus, in every slot of the CFP the system imitates the behavior of a single AP([13]–[15]), and therefore guarantees fairness and QoS. For determining the slot assignment, we designed an efficient scheduling algorithm with proven guarantees on its performance. The algorithm ensures that the number of slots allocated to each access point is proportional to its load and moreover it maximizes each AP's share of time when it is activated. In the CFP, the system ensures fairness and QoS to both RT and non RT sessions by providing service to backlogged sessions. The CP is also used for data transmission and as a signaling channel for management messages and for initiating new sessions. By regulating the proportion of time spent in the CFP versus the CP, the system is able to trade off the achieved fairness with the overall network throughput. Our scheme requires modifications only at the access points and it can be implemented without requiring any changes to 802.11 MAC layer standard. We do not require any additional software or any changes in the mobile stations. Moreover, the scheme can be viewed as a complementary mechanism for enhancing the 802.11-E proposal [18]. By extensive simulations we show that the system indeed provides fair service to its users and can support RT-sessions even in large networks.

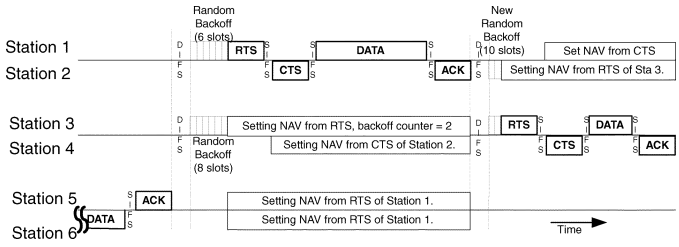

Fig. 1. An example of station transmissions in DCF mode.

II. The IEEE 802.11 Standard

A. The MAC Layer of the IEEE 802.11 Standard

We now provide a brief description of the two coordination functions of the IEEE 802.11 MAC protocol [18]. We consider only WLANs that operate in *infrastructure mode* and contain multiple *access points* (APs).

1) The Distributed Coordination Function (DCF): The DCF is a distributed medium access method that employs the carrier sense multiple access/collision avoidance (CSMA/CA) mechanism. Each station senses the wireless channel before transmission by using both physical and virtual carrier sensing mechanisms, and it sends its message only if the channel is idle at least for a period of DCF interframe spacing (DIFS). At the end of each successful transmission, the receiving station acknowledges the reception by sending an ACK message after a short interframe spacing (SIFS). In this approach, collisions occurs as a result of several stations detecting idle channel at the same time and initiating simultaneous transmissions.

For reducing the interferences that result from the hidden node problem (see Section II-B for details), the standard defines an optional request-to-send/clear-to-send (RTS/CTS) handshake. A station initiates its transmission by sending a short RTS message that contains the message destination and a *duration field*. The latter defines the required period for data transmission and the CTS and ACK replies. Upon reception of a RTS message, the destination replies after a SIFS with a CTS message that also carries a duration field. Each station that decodes the RTS or CTS message, sets an internal timer, termed the *network allocation vector* (NAV), for this period and defers its transmission appropriately. An example of a DCF message transmission is illustrated in Fig. 1. Recall that a station may sense a busy channel but may not be able to decode the message as a result of a collision or due to low signal-to-interference ratio (SIR). In such a case, the station defers its transmission for a period of at least extended interframe spacing (EIFS) which is much longer than the DIFS period.

2) The Point Coordination Function (PCF): The PCF is used for supporting delay sensitive applications. In this mode, time is divided into repeated periods that are called *superframes*. Each superframe contains both a *contention free period* (CFP) managed by the PCF and a *contention period* (CP) in which the DCF is used for accessing the channel. For starting a CFP, the AP transmits a beacon frame for all the stations in its transmission range to enter the PCF mode. During the CFP, the AP polls its associated stations according to a predetermined order, called *polling list*. No station is allowed to transmit unless it is polled and it enables the AP to receive and deliver messages without interference. The CFP ends when the AP sends a CF-end message.

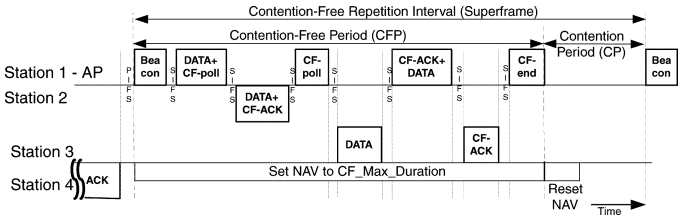

Fig. 2. An example of station transmissions in PCF mode.

An example of a superframe is depicted in Fig. 2. Currently, the PCF mode is an optional part of the standard. However, the IEEE 802.11-E QoS Extension [18] contains a mandatory contention free polling mechanism that broadens the PCF mode. In this work, we assume that all stations support the PCF mode.

B. The Fairness and QoS Limitations of the Standard

Generally speaking, fairness is the networks ability to provide the same service level to all its users during a given time internal. This property is essential for providing QoS assurance for real-time applications, usually given in terms of guaranteed bandwidth, delay bound and jitter. However, these guarantees cannot be supported in the current 802.11 standard. The DCF is designed to support only best-effort services and the distributed MAC protocol cannot provide any guarantee on message delays. The PCF mode provides certain degree of fairness in the case of a single AP, however it cannot support the required guarantees in networks with several APs. In such networks, transmission collisions may occur during a CFP as a result of the hidden node and overlapping cell problems [16]. The *overlapping cell problem* refers to the issue of interference during a CFP resulting from the transmissions in adjacent cells and it is a special case of a more subtle *hidden node* problem. A station is called *hidden* when it is in the sensing range of the intended receiver but out of the sensing range of the transmitter. Thus, a transmission by a hidden station may prevent a receiver from decoding a message intended for it. The 802.11 standard addresses this problem by using both physical and virtual sensing mechanisms (the latter is supported by the RTS/CTS-handshake). However, these mechanisms provide only partial solution to these problems.

The hidden node problem causes a significant reduction in the system throughput and prevents the APs from providing fair service to their users even during their CFPs. In this case, appropriate synchronization of the CFPs is not sufficient, since the hidden node may be a mobile user operating in DCF mode that has not received the beacon message at the beginning of a CFP. Our simulations indicate that the affect of hidden nodes is more severe on users near the boundaries of an AP's transmission range, as illustrated in Fig. 3. (The details of our simulations can be found in Section VIII.) For such users, the signal from their associated AP is very weak, while the aggregated noise from adjacent cells is elevated, thus making them highly prone to interference. Moreover, the backoff mechanism of the DCF mode itself may have a detrimental effect on the service level obtained by these users, as the contention window is doubled after each failed transmission. Consequently, the service level that a mobile user experiences depends on its distance from its associated AP. While closer users benefit from high throughput and low delays, remote users are actually starved.

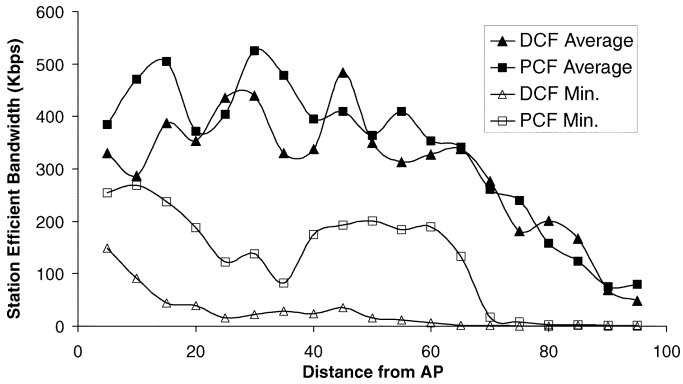

Fig. 3. The provided service level of the DCF and PCF modes with frequency planning, where the transmission range of an AP is 100.

C. The IEEE 802.11-E Proposal

Presently, a significant effort is underway, by the IEEE 802.11 standard committee, to add QoS assurance capabilities to the current standard. This committee is working on a QoS enhancement scheme for the 802.11 MAC protocol, termed IEEE 802.11-E, [18]. The proposal is based on improving the current DCF and PCF operating modes. It suggests a distributed priority scheme for the DCF mode for providing priority access to RT traffic over "best-effort" traffic. In addition, the proposal contains a mandatory hybrid coordination function (HCF) that provides a contention free polling mechanism and extends the current PCF mode.

Although the proposal is designed for adding QoS capabilities to the standard, it does not explicitly address the hidden node and the overlapping cell problems. Therefore, it still suffers from the same spatial unfairness problem described before. Since our scheme addresses this problem by synchronizing the CFPs, it can be viewed as a complementary mechanism for enhancing the 802.11-E proposal.

III. The Network and the Channel Models

We consider a WLAN operating in the infrastructure mode and supporting a large number of APs. All the APs are attached to a single distribution system (DS) that connects them via gateways to wired data and voice networks such as the Internet. The network provides connectivity services to the mobile users that reside in its coverage area, comprised of the APs transmission ranges. At any given time, each mobile user is associated with a single AP.

Since IEEE 802.11 based WLANs use multiple communication bit-rates, we allocate to each mobile user u an integer weight w_u, which is inversely proportional to its communication bit-rate with its associated AP. A user weight reflects the time period required for sending (or receiving) a message of a given length to (or from) this user. For instance, in 802.11-b networks we allocate weights of 2, 4, 11, and 22 for users with communication bit-rates of 11 Mbps, 5.5 Mbps, 2 Mbps, and 1 Mbps, respectively. Moreover, to each AP $v \in V$, we assign an *accumulated user weight*, or simple a *weight*, denoted by m_v which is the total weight of all the mobile users that are associated with the AP v. In our model we do not consider a specific message length. Practically, the accumulated user weight represents the amount of time that is required by the AP to send a message of given length to each one of its associated user. We

define as a *time unit* (TU) the required time period for transmitting the shortest message with the maximal allowed bit-rate and we use this unit as a metric for specifying the transmission time of messages. We also assume a known upper bound on message transmission time.

We consider two types of mobile users, *real-time* (RT) and *non-real-time* (NRT) users. While NRT-users are sporadically involved in data delivery, RT-users are engaged in traffic transmission only during RT-sessions, i.e., voice calls, that provide QoS guarantees for the session bandwidth and packet delay. For initiating a RT-session, an RT-user sends a request to its associated AP, and the latter determines whether to accept or reject the request. For our analysis, we assume that NRT-users always have pending messages to send or receive, while RT-users are occasionally engaged in a RT-session, i.e., voice calls. Each RT-session requires a constant bit rate (CBR) and an RT-user needs to either receive or send a RT-message every D time units. In the following we use the term *stations* for denoting both APs and mobile users.

We now turn to describe the main characteristics of the wireless channel. The IEEE 802.11 standard allows us to use only limited number of non-interfering channels. In particular, the IEEE 802.11 b/g standards provide only three non-interfering channels. Each AP and its associated mobile users utilize only a single channel and use it as a shared medium. As a result, simultaneous transmissions may interfere with each other. The ability of a station to decode a message depends only on the signal-to-interference ratio (SIR) defined by the strength of the received signal over the accumulated strength of the other interfering transmissions. Clearly, the interfering signal may be a combination of several simultaneous transmissions, however, due to the carrier sensing mechanisms in the IEEE 802.11 standard, we assume that also in these cases, there is a single transmission which is the dominant contributor to an interfering signal. Consequently, we say that two APs are *interfering* when a message exchanged by one AP may prevent a proper message decoding in the vicinity of the other AP.

The scheme presented in this study can be deployed in practical multi-rate 802.11 networks without any modification of the standard or the mobile user behavior. However, for the analysis of our approximation algorithms, we assume that the main factor that affects signal attenuation is the path loss and we ignore secondary effects like multipath and shadowing [19]. Consequently, we associate each station v with two *circular regions*. The first is its *transmission range* that defines the zone in which any message sent by station v can be correctly decoded and its radius is R_T. The second is the station *sensing range* such that any station included in this range can sense every transmission of v. The radius of the sensing range is R_T and in practice $R_S > 2 \cdot R_T$.

IV. The System Goals

In the following, we use the following definition. We say the *efficient bandwidth* B_v of a station v, also termed the station *flow*, is the amount of data that station v transmits or receives successfully in a given time period. Thus, the efficient bandwidth of an AP is the aggregated flow that the AP successfully exchanges with its associated users and the networks throughput is the total flow of all its APs.

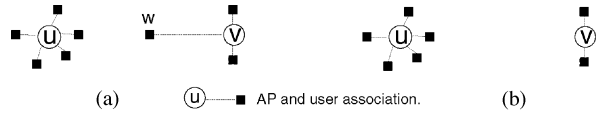

Fig. 4. The relationship between fairness and throughput.

The main goal of our system is to provide fair service to all its mobile users and to ensure QoS guarantees to real-time sessions, while maximizing the achievable overall network throughput. Intuitively, a system provides a fair service if every user experiences the same network usage and has the same flow as any other user. This notion of fairness assumes that all users are identical and have the same requirements. However, in our system, RT users with very strict QoS requirements for latency and bandwidth co-exist with NRT users that would only like to maximize their average throughput. Thus, *our notion of fairness is to ensure that users of a single type experience the same network usage and the network resources are proportionally allocated among the users of the two types.* Moreover, the experienced service level should be independent of the distance between the users and their associated APs. In other words, we require *spatial fairness*. To achieve our fairness goals we need to consider both *inter-AP* and *intra-AP* fairness. An AP provides *intra-AP fairness* by balancing between resource allocation to its NRT and RT-users according to a given *fairness criteria*. For instance, a plausible fairness criteria may require that the success probability of a RT-session request be proportional to the efficient bandwidth given to each NRT-user. In addition, an *inter-AP* fairness is obtained when the efficient bandwidth B_v of an AP v is directly proportional to its weight m_v, defined as the aggregated weight of its associated users. Thus, we say the network is *inter-AP fair* if the ratio B_v/m_v is the same for all APs. In wireless networks achieving intra-AP or inter-AP fairness is contradictory to achieving high throughput, as illustrated by Examples 1 and 2.

Example 1: Consider two APs u, v, as depicted in Fig. 4(a). Here user w is associated with AP v. Due to the large distance between the two APs, both of them can simultaneously exchange messages with their adjacent users (except user w). However, when v is exchanging messages with user w then u and all its associated users must be silent. Thus, the network throughput is maximized by starving user w, i.e., in order to maximize the network throughput the intra-AP fairness (spatial fairness) requirement must be violated. ■

Example 1 illustrates that inter-AP fairness requires coordinating the operations of adjacent APs, resulting in a reduction of the overall network throughput.

Example 2: In Fig. 4(b), the communication of the two APs does not interfere and both APs have the same efficient bandwidth (assuming $B_v = B_u = 1$ Mbps). In this scenario, inter-AP fairness requires that all the users in the system experience the same flow allocation, 1/5, since AP u has five associated users. However, the network utilization can be increased by allowing increased flow allocation of 1/2, to the users associated with AP v without affecting the flow allocations of the users attached to AP u. ■

We resolve such issues as illustrated in Example 2, by considering a *maxmin fairness* requirement that seeks to find a flow allocation that maximizes the ratio $\min B_v/m_v$ for all the APs v. After maximizing this ratio, our scheme maximizes the system overall throughput by allocating the residual efficient bandwidth of the network to the different APs.

Examples 1 and 2 illustrate the need for a system that would help the network administrator to strike a balance between fairness and throughput. We show later how MiFi achieves these goals.

V. An Overview of the MiFi System

In this section, we present an overview of the MiFi system. The system contains a *network operation center* (NOC) that coordinates the APs. For management purpose, we require the internal clocks of the NOC and the APs to be synchronized to a certain degree of accuracy.[1] However, for ease of presentation we henceforth assume that the clock gap between any pair of APs is negligible. On-board each AP, a special software is used to control its behavior for providing QoS and fairness to the attached users and for communicating with the NOC. However, the MiFi system does not require any modification of the IEEE 802.11 standard or the software of the mobile users. We only assume that mobile users are able to convey, via request messages, the type of session they wish to initiate, which can be either a RT or a NRT session.

A. General Description

Our scheme is inspired by the recent results of [13], and [14]. These papers have shown that the PCF mode can efficiently support real-time sessions and provide fairness in WLAN when all the users are in the transmission range of a *single* AP. In this work, we extend this result also to networks with *multiple* APs. This challenging goal cannot be obtained easily, mainly due to the overlapping cell and the hidden node problems. For achieving this goal, the MiFi system imitates the behavior of a single AP. We partition time into repeated periods or *superframes*. Each superframe has a fixed duration D and it contains a *contention free period* (CFP) followed by a *contention period* (CP). The CFP is used for data transmissions of both RT and NRT sessions, while ensuring inter-AP fairness and QoS support. Thus, the CP is mainly used as a signaling channel for initiating new sessions and sending management messages. The proportion of time allocated to each period is determined by the system needs to balance between fairness and network throughput. The CFP starts with a *beacon block* (BB) in which all the APs transmit "almost" at the same time beacon messages for initiating a CFP in their vicinity. It ends with an *end block* (EB) in which all the APs send CF-end messages approximately at the same time to end their CFPs. This is illustrated in Fig. 5. During the CFP, the APs poll their associated users according to their *polling lists*, and only stations that are polled are allowed to transmit. *By ensuring that the CFPs of all the APs start and end at the same time, our scheme solves the hidden node problem that results from spontaneous transmissions of mobile users operating in the DCF mode.*

[1]The synchronization of the APs and the NOC can be obtained since they are all connected to the same local network.

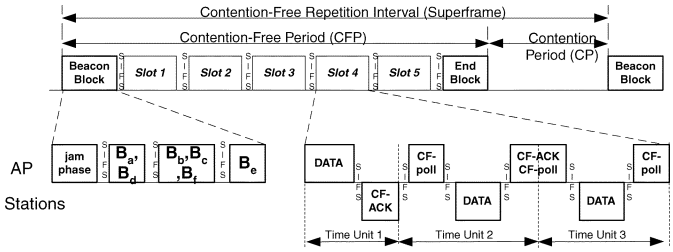

Fig. 5. A MiFi superframe with five slots, each slot three time units long.

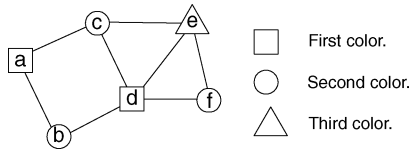

Fig. 6. The interference graph of a WLAN with six APs.

Our scheme also overcomes the overlapping cell problem as follows. We divide the CFP into slots, each of size at least Δ time units, for a specified parameter Δ. The slots are efficiently allocated by the NOC to the APs such that no two APs whose transmissions may interfere get the same slot. Every AP is allowed to poll its users only during its allocated slots. In addition, it is ensured that transmissions in a slot terminate by the end of the slot, thus avoiding collisions with transmissions in the following slots. In our system the NOC only determines the slot assignments and synchronizes the APs. Each AP manages its own admission control mechanism for accepting new RT-sessions and determines its own order for polling its users. An example of the CFP partitioning into slots and the polling mechanism is described in Fig. 5. For simplicity, we assume in our examples that every message transmission captures only one time unit. We now provide more details of our scheme.

B. The Interference Graph

We say that two APs are *interfering* when a message exchanged by one AP may prevent a proper message decoding in the vicinity of the other AP. The MiFi system overcomes the overlapping cell problem, by allocating disjoint sets of slots to interfering APs so that only non-interfering APs may use the same slot. These interference relationships are represented by an *interference graph*, defined as follows.

Definition 1 (Interference Graph): The *interference graph* $G(V, E)$ is defined by the set V of APs and a set of edges E between every pair of interfering APs $u, v \in V$.

In the case of the circular region channel model, as defined in Section III, it is easy to show that:

Property 1: The distance between any pair of interfering APs is at most $2 \cdot R_T + Rs$.

Note that even if two APs are less than $2 \cdot R_T + Rs$ apart they may not be interfering. However, we attempt to provide "hard" QoS guarantees by assuming that two APs are interfering if their distance is at most $2 \cdot R_T + Rs$. Property 1 implies that in circular region channel model the interference graph is actually a unit disk graph [20]. An example of an interference graph $G(V, E)$ is depicted in Fig. 6. In Section VII, we elaborate on our methods for detecting the interference graph when the APs' locations are not known.

The achievable throughput of the MiFi system depends on the density of the interfering graph G. Generally speaking, the

number of slots allocated to each AP is inversely proportional to its degree in G. In practice, we may eliminate from G the edges whose endpoints have low probability of interfering with each other, and still get adequate QoS guarantees. We leave for a future work the problem of constructing an interfering graph that allows the MiFi system to provide sufficient QoS assurance while exploiting spatial reuse more efficiently.

C. The Beacon and the End Blocks

In the 802.11 standard, each CFP starts with a beacon message and ends with a CF-end message. In our scheme, we ensure that every mobile user successfully receives these messages from its associated AP by the usage of beacon blocks and end blocks. For simplicity, we describe only the beacon block (BB). The end block (EB) has similar description. Recall that a station in DCF mode initiates a transmission by sending a RTS packet with a duration field. A station, that senses a signal, which it is unable to decode, postpones its transmission for a period of EIFS. The BB contains two phases, a *jamming* phase followed by a *beacon transmission* phase. The jamming phase silences the network for a period of EIFS, so that no users are allowed to transmit. In the beacon transmission phase, APs send their beacon messages. Since the users are silent in the beacon transmission phase the beacon messages do not suffer from collisions with messages transmitted by mobile users operating in DCF mode.

Let us assume, for illustration, that the beacon block starts at time t_0. Any RTS messages that originate before time t_0 and whose data and ACK transmissions are supposed to end after time t_0 are ignored by the APs. Thus, at time t_0 there are no transmissions of data or ACK messages in the network and the only possible transmissions are for RTS messages. At time t_0, all the APs start to jam the channel, which is done for a period of $RTS\_TIME + DIFS$, where RTS_TIME is the time required for sending a RTS message at the lowest bit rate.[2] As a result, all the mobile users, including those that have sent RTS messages at time t_0, sense the jammed signal and set their NAV to EIFS. At the end of the jamming phase the APs send their beacon messages in the beacon transmission phase. Since beacon messages from two interfering APs may collide, the beacon transmissions of APs are synchronized such that two adjacent APs in the interference graph do not send their beacon messages simultaneously. For reducing the overhead of the beacon block, we would like to send the beacon messages as quickly as possible. Thus, we map the beacon synchronization problem into a graph coloring problem that seeks to find the minimal number of colors that are needed to color the interference graph, such that all the nodes with the same color send their beacon messages simultaneously, as described in Example 3. The details of the coloring algorithm are given in Section VI. Note that the beacon transmission phase may be required to be longer than an EIFS period. In this case, after several beacon transmissions, the APs simultaneously transmit a short jamming signal, for resetting the NAV of the mobile users to an additional EIFS period.

Example 3: An example of a beacon block is shown in Fig. 5 for the interference graph $G(V, E)$ presented in Fig. 6. Since $G(V, E)$ is 3-colorable, the beacon block contains three *beacon slots*. In the first slot nodes a and d transmit their beacon messages, in the second slot a beacon is sent by nodes b, c and f, and finally in the third slot node e sends a beacon message. ■

[2]In networks where the RTS/CTS handshake is not utilized, the duration of the jamming phase should be slightly more than the maximal duration of a message transmission.

D. The Slot Assignment Mechanism

As described above, all the APs share a common CFP that is partitioned into slots, each of size at least Δ time units. Slots are assigned to APs such that no two interfering APs get scheduled in the same slot. Thus, the goal of the *slot assignment mechanism* is to maximize the network throughput while ensuring inter-AP fairness.

Let the CFP be divided into R slots enumerated from 1 to R. We denote by S_v the set of slots that are allocated to AP v and by r_v the number of slots in S_v. A *slot assignment* is a vector $\mathcal{S} = \{S_{v_1}, S_{v_2}, \ldots, S_{v_{|V|}}\}$, of the sets S_{v_i} for every AP $v_i \in V$. A slots assignment is termed *feasible* if for every AP v, $S_v \subseteq [1 \ldots R]$ and any pair of adjacent nodes in the interference graph $G(V, E)$ do not have any common slot, i.e., for every $(u, v) \in E$ it follows that $S_u \bigcap S_v = \emptyset$. For obtaining both inter-AP fairness and high throughput, we say that a feasible slot assignment $\mathcal{S}$ is *optimal* if it maximizes the *min-slots-to-weight ratio* defined by $\rho = \min_{v \in V}(r_v/m_v)$. We show, in Section VI-C, that the problem of finding the optimal slot assignment is NP-hard and we present efficient approximation algorithms for the problem.

Our slot assignment scheme has two components. The first is a *coloring algorithm* that given a graph $G(V, E)$ and the number of colors, r_v, required by every node $v \in V$, finds a feasible color assignment with minimal (within a factor 3) number of colors. The second component uses the coloring scheme as a building block for finding an efficient slot assignment. It performs a binary search for finding the maximal min-slots-to-weight ratio ρ that requires no more than R slots. At each iteration, it selects a ratio ρ and sets the requirement of every node $v \in V$ to $r_v = \lceil \rho \cdot m_v \rceil$ colors. The algorithm then uses the coloring algorithm to check whether or not there is a feasible slot assignment with R slots (colors). Based on the result, the algorithm picks lower or higher value for the ratio ρ until it quickly converges to the optimal ratio ρ. The details of slot assignment algorithm and its approximation analysis are given in Section VI-C.

Example 4: In Fig. 7(a), we present a slot assignment that maximizes the min-slots-to-weight ratio of the given interfering graph $G(V, E)$. Each superframe contains five slots and the weight m_v of each AP $v \in V$ is depicted near each node v. The figure shows the allocated slots S_v as well as the slots-to-weight ratio $|S_v|/m_v$ of each node v. In this case the maximal min-slots-to-weight ratio is 1/5 due to nodes c and e. ■

Here, we consider further enhancements to the slot assignment mechanism for improving the system throughput. Consider a slot j and let V_j be the set of APs that are allowed to use slot j by the slot assignment algorithm. The network utilization is maximized during the period of slot j only if the set V_j is a *dominating set* of the interfering graph $G(V, E)$, i.e., every node $v \in V$ is either included in V_j or is adjacent to a node in V_j. Note that if V_j is not a dominating set then there is an AP $u \notin V_j$ that is not adjacent to any other AP in V_j. Thus, node u may use slot j without interfering with the APs in V_j, resulting in an increase in the network utilization. Consequently,

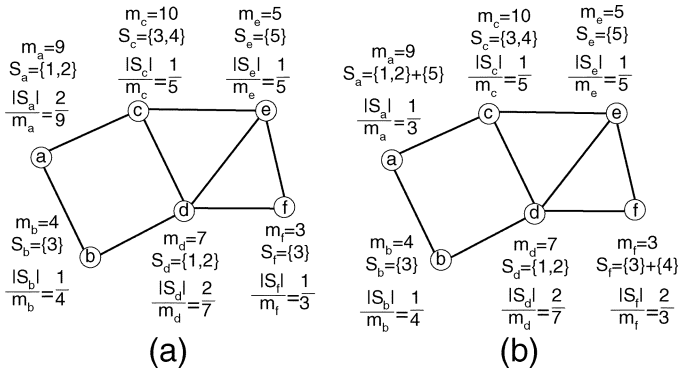

Fig. 7. An example slot assignment for maximizing the min-slots-to-weight ratio.

for maximizing the network throughput at each slot j without causing interferences, we seek a dominating independent set V_j' that includes all the nodes in the set V_j. We calculate the set V_j' by constructing a unit disk subgraph $\tilde{G}(\tilde{V}, \tilde{E})$ of graph G. The nodes $\tilde{V} \subset V$ are the ones that are not included in V_j and are not adjacent to any node in V_j in G. We use the algorithm in [21] for finding a maximal independent set of $\tilde{G}$, denoted by $\tilde{V}'$. The nodes then assigned to slot j are $V_j' = V_j \bigcup \tilde{V}'$. Recall that the set V_j' is a dominating independent set of G and therefore the throughput of the network is increased without violating the inter-AP fairness, as illustrated in Example 5.

Example 5: In the slot assignment of Example 4, the sets V_4 and V_5 defined by the colors 4 and 5, respectively, are not dominating sets. These sets can be augmented with additional nodes while keeping the independent set property. After this augmentation, the new set V_4' also includes node f and node a is included in the set V_5'. Consequently, the network flow is increased by 22%, from 9 to 11 allocated nodes per superframe, as depicted in Fig. 7(b). ■

The 802.11 standard allows us to use three non-overlapping frequencies F for reducing interference and thus increasing the network throughput. Every AP is associated with a frequency $f \in F$ and a slot is identified by a pair (f, c) of a frequency $f \in [1 \ldots |F|]$ and a color $c \in [1 \ldots R]$. Here, a slot assignment is feasible only if all the slots allocated to an AP belong to the same frequency but have different colors and any two adjacent APs do not share the same (f, c) pair. In Section VI-D, we present an efficient slot assignment algorithm for the case when there are multiple non-overlapping frequencies.

E. The Admission Control and the Polling List

In a MiFi system, each AP needs to maximize its efficient bandwidth while providing fair service to its RT and NRT users. For ensuring intra-AP fairness, each AP employs an *admission control* mechanism that enforces a given *fairness criteria*. For instance, consider an AP v that has r_v slots and its weight is m_v. Moreover, let m_v^{RT} be the aggregated weights of all its associated RT-users and let Δ be the number of time units in every slots. Our admission control approves new RT-session request only while the aggregated flow allocated for RT-users does not exceed a threshold of $H_v = c \cdot m_v^{RT} / m_v \cdot r_v \cdot \Delta$, for a given configuration parameter $c \geq 1$ and a requirement that $H_v < r_v \cdot \Delta$. Such admission control balances between the success probability of RT-session requests versus the average flow given to each NRT-user. Other mechanisms can be found in [14], [13].

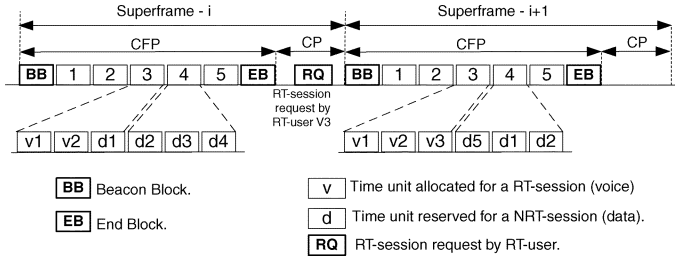

Fig. 8. An example of a polling list of node c.

An RT-user initiates a new RT-session by sending a request to its AP during the CP. If the AP approves the request then it allocates a time unit to this user and adds the users address to its polling list. In the CFP the AP first polls all the RT-users with active RT-sessions and in the remaining time of its slots it polls its NRT-users. Since H_v is smaller than the time units available to AP v, i.e., $r_v \cdot \Delta$, every RT-user engaged in an active RT-session is either polled or receives data at each superframe. For ensuring intra-AP fairness, the AP employs a sliding window method for determining the next NRT-user to poll at time t. The AP keeps records of the number of successfully served messages (either sent or received) by each NRT-user during a time period of $[t - T, t]$, for a sliding window of size T. The polled NRT-user is the one that has experienced the lowest bandwidth during the window period and its messages can be accommodated in the remaining slot time. Note that the intra-AP fairness is improved by increasing the size of the sliding window, T. An illustration of the polling mechanism is given in Fig. 8.

The main inefficiency of the polling mechanism may result form unsuccessful polling attempts. In practice, NRT-stations do not always have packets to send and polling these stations decreases the system utilization. Our method for reducing this number is based on the following observations. Most of the traffic is from the APs to the mobile-users, i.e., web browsing and e-mail. Moreover, the majority of the packets are originated as a response to a message reception, like TCP acknowledgment. In our approach, the AP polls a user after sending it a packet and the users use the CP for sending or initiating sessions or for resuming their operations. The latter may be sent with high priority, as proposed by the IEEE 802.11-E proposal [5], for expediting the session initiations.

F. Mobility Management

User mobility does not raise any major concern for the system operation and users may change their association as they move. However, user mobility may reduce the inter-AP fairness since it affectsthe APs' slots-to-weight ratios. To this end, the NOC periodically re-calculates the current slots-to-weight ratio and compares it with the best possible ratio. If the gap between the two ratios is significant, then the slot assignment is modified.

VI. The Slot Assignment and the Frequency Planning Algorithms

We now describe our algorithms for the slot assignment problem for the two cases when we have a single frequency and when we have multiple frequencies. First we present two relevant graph coloring problems both of which are NP-hard, and we present efficient approximation algorithms for solving them. It is already known that it is even hard to find an approximate solution for the graph coloring problem, i.e., the

problem cannot be approximated within $|V|^{1-\epsilon}$ for any $\epsilon > 0$, unless $Z_{PP} = NP$, [22]. Thus, although our algorithms can be used for general graphs, we limited our analysis only to the case of circular region wireless channel, where the induced interference graph is a unit disk graph [23]. These algorithms are then used as a building block for constructing solutions for the single and multiple frequency slot assignment problems.

A. The Graph Coloring and Frequency Assignment Problem

Let $G(V, E)$ be the interference graph defined in Section V-B for circular region wireless channel model and recall that G is a unit disk graph. We first define a coloring problem for G that is relevant for our purposes. We assume that each node $v \in V$ is associated with an integer requirement $r_v \geq 1$, which stands for the number of distinct colors required by node v. We define the coloring problem for G as an assignment of r_v distinct colors S_v to every node $v \in V$ such that no common color is assigned to the two end nodes of any edge $(u, v) \in E$, i.e., $S_u \cap S_v = \emptyset$ and such that the total number of colors used $|\bigcup_{v \in V} S_v|$ is minimized. Next, we define a joint coloring and frequency assignment problem for G that is relevant for our purposes. Here in addition to finding a coloring for nodes of G we need to assign frequencies from a given set F to the nodes of G, such that each node v is assigned a single frequency $f_v \in F$ and such that $S_u \cap S_v = \emptyset$ for only those edges $(u, v) \in E$ such that $f_u = f_v$. The objective is still to minimize the total number of colors used: $|\bigcup_{v \in V} S_v|$.

It was shown in [23] that the problem of deciding whether a unit disk graph with unit requirements and $|F| = 1$ can be colored with three colors is NP-complete. This implies that the joint coloring and frequency assignment problem as defined above cannot be approximated to a ratio 4/3 or better unless $P = NP$. It was also shown in [24] that deciding whether a unit disk graph with unit requirements and $|F| = 1$ can be colored with k colors is also NP-complete for all $k \geq 3$. Note that this implies that the more general coloring and frequency assignment problem is also NP-complete for $k \geq 3$. On a positive side it was shown in [25] and [20] that there is a 3-approximation algorithm for the unit disk graph coloring problem with unit requirements and $|F| = 1$. In addition these results imply, as described later, a 3-approximation algorithm for the unit disk graph coloring problem with general requirements and $|F| = 1$.

B. The Coloring Algorithm for a Single Frequency

Here we assume that $|F| = 1$. Note that in this case the joint coloring and frequency problem reduces to the coloring problem, without regards to frequency assignment. As mentioned in Section VI-A, the result of [20] and [25] implies a 3-approximation algorithm for the unit disk coloring problem with general requirements. We sketch a 3-approximation algorithm for completeness and also for presenting the ideas that we will use for designing an approximation algorithm for the more general coloring and frequency assignment problem.

We first present a 3-approximation algorithm with the assumption that the location of the unit disks are known. Later we show how to extend it to the case when the locations may not be known. We denote by $N(v)$ the set of neighbors of node v in the graph G. We assume that the colors are numbered 1, 2, 3,..., etc. Let $v_1, v_2, \ldots, v_n$ be the nodes of G ordered in non-decreasing X-coordinate of the center of their unit disks. The first coloring algorithm A works as follows:

```
Alg. A
    For i = n down to 1
        Assign r_{v_i} minimum available colors to v_i
    End For
```

Algorithm A uses First-Fit for its color assignment: namely it chooses to assign the minimum available colors to the nodes, where the set of available colors for a node are those colors that have not been assigned to any of its neighbor. More formally the set of colors assigned by the algorithm to the nodes of the network are determined as follows. Let S_v denote the set of colors assigned to node v by algorithm A. Let $N'(v_i) \subseteq N(v_i)$ be the set of neighbors of node v_i among the nodes $v_{i+1}, v_{i+2}, \ldots, v_n$. Thus, $N'(v_n) = \emptyset$. Let $S(N'(v_i)) = \bigcup_{v \in N'(v_i)} S_v$ be the set of distinct colors assigned by A to the nodes in the set $N'(v_i)$. Let Z_i be the set of all positive integers other than those in $S(N'(v_i))$. Then Z_i is the set of available colors for node v_i and A sets S_{v_i} to the least r_{v_i} colors in Z_i.

Example 6: Consider the graph G shown in Fig. 6. Here nodes a, c and d have requirements $r_v = 2$, while the rest of the nodes have requirement 1. The nodes of G when ordered in non-decreasing X-coordinate of the center of their unit disks yields the sequence a, b, c, d, e, f. Thus, algorithm A first assigns color 1 to node f. Next, it assigns color 2 to node e. Note that at this point the set of available colors for node d are all colors except 1 and 2. Hence, A assigns colors 3 and 4 to node d. Next, algorithm A assigns two colors to node c and the minimum available colors are 1 and 5. Node b is next and is assigned color 1. Finally, node a is assigned colors 2 and 3. Thus, on graph G algorithm A uses 5 colors which is the minimum number of colors required by any algorithm for graph G. ■

We now show that A is a 3-approximation algorithm for coloring unit disk graphs.

Claim 1: The largest color in S_{v_i} is at most $\sum_{u \in N'(v_i)} r_u + r_{v_i}$.

Proof: Note that the number of distinct color assigned by any coloring algorithm to the set of nodes $N'(v_i)$ is at most $\sum_{u \in N'(v_i)} r_u$. Thus, the set of available colors for node v_i include all but at most $\sum_{u \in N'(v_i)} r_u$ colors. Thus, the largest color in S_{v_i} is maximized when the colors in $S(N'(v_i))$ are $1, 2, \ldots, \sum_{u \in N'(v_i)} r_u$. Thus, the r_{v_i}th smallest color in Z_i is at most $\sum_{u \in N'(v_i)} r_u + r_{v_i}$. ■

Claim 2: The number of colors used by A is at most $\max_i \{\sum_{u \in N'(v_i)} r_u + r_{v_i}\}$.

Proof: Let A use k colors. Let v_j be a node in G that is assigned color k by A. Then by Claim 1 we have $k \leq \sum_{u \in N'(v_j)} r_u + r_{v_j}$. Hence, the result follows. ■

Claim 3: The minimal number of required colors is $\max_i \lceil (\sum_{u \in N'(v_i)} r_u + r_{v_i})/3 \rceil$.

Proof: Let G be a unit disk graph. Consider a node v of G the center of whose unit disk has the smallest (ties broken arbitrarily) X-coordinate. It was shown in [20] that the set of nodes $\{v\} \cup N(v)$ does not contain an independent set of size more than 3 in G. Thus, the nodes $\{v\} \cup N(v)$ require at least $\lceil (\sum_{u \in N(v)} r_u + r_v)/3 \rceil$ distinct colors in any feasible coloring of G, since the nodes assigned a particular color must form an independent set of G. Now consider the unit disk graph consisting of only the nodes $v_i, v_{i+1}, \ldots, v_n$. In this graph v_i is a node the center of whose unit disk has the smallest X-coordinate and in this graph the set of neighbors of v_i are the nodes in

the set $N'(v_i)$. Thus, the nodes in $\{v_i\} \cup N'(v_i)$ require at least $\lceil (\sum_{u \in N'(v_i)} r_u + r_{v_i})/3 \rceil$ distinct colors. Taking the maximum over all nodes v_i we get the claimed result. ■

Corollary 1: Claim 2 and Claim 3 imply that *A* is a 3-approximation algorithm for coloring unit disk graphs.

Next, we present a unit disk coloring algorithm *B* that can be used even when the disk locations are unknown. Consider an ordering of the vertices $v_1, v_2, \ldots, v_n$ of G which is obtained as follows. Let $G_1 = G$. Let $N^1(v) \subseteq N(v)$ be the set of neighbors of node v in G_1. Let v_1 be a node that minimizes $\sum_{u \in N^1(v_i)} r_u + r_{v_i}$ for all vertices v_i in G_1 (ties broken arbitrarily). Let G_2 be the graph obtained by removing the unit disk corresponding to node v_1 from G_1. Let $N^2(v) \subseteq N(v)$ be the set of neighbors of node v in G_2 and let v_2 be a node in G_2 that minimizes $\sum_{u \in N^2(v_i)} r_u + r_{v_i}$ for all node v_i in G_2. v_3 and $N^3(v)$ are similarly defined for G_3 obtained from G_2 by deleting node v_2 and so on. The coloring algorithm *B* also uses First-Fit just like coloring algorithm *A* (but with a potentially different ordering of vertices):

Alg. *B*
For $i = n$ down to 1
Assign r_{v_i} minimum available colors to v_i
End For

Example 7: Consider the graph G shown in Fig. 6. Consider $G_1 = G$. In it the total requirement $\sum_{u \in N^1(v)} r_u + r_v$ of node v and its neighbors is as follows: For node f the total requirement is 4. For nodes b and a the total requirement is 5. For nodes c and d the total requirement is 7, and for node e it is 6, as illustrated in Fig. 9. Algorithm *B* picks $v_1 = f$. In graph G_2, obtained by removing v_1 from G_1, the total requirements of nodes d and e decrease by 1 to 6 and 5, respectively, while for the other nodes the total requirement stays the same. Algorithm *B* sets node $v_2 = b$. Thus, in G_3 the total requirement of nodes a and d decrease by 1 to 4 and 5 correspondingly. In graph G_4, obtained by removing node $v_3 = a$, only node c decreases by 2 to 5 so the total requirement of each one of the remaining nodes is 5. Similarly, in the following, Algorithm *B* sets node $v_4 = c$, $v_5 = d$ and $v_6 = e$. Thus, the ordering of the nodes of G by algorithm *B* is f, b, a, c, d, e. Therefore, algorithm *B* first assigns color 1 to node e. Next, it assigns color 2 and 3 to node b and then it assign the colors 4 and 5 to node c, as shown in Fig. 9(g). Note that at this point the set of available colors for node a are all colors except 4 and 5. Hence, *B* assigns colors 1 and 2 to node a. Next, algorithm *B* assigns a single color to node b. Since $S_a = \{1, 2\}$ and $S_d = \{2, 3\}$, the minimum available color for node b is 4. Finally, node f is assigned color 4. Thus, on graph G algorithm *B* uses 5 colors which is the minimum number of colors required by any algorithm for graph G. ■

Claim 4: *B* is also a 3-approximation algorithm for the problem of coloring unit disk graphs

Proof: Let v'_1 be a node whose unit disk's center has the minimum X-coordinate among the unit disks in G_1. Let v'_2 be a node whose unit disk's center has the minimum X-coordinate among the unit disks in G_2. v'_3 is similarly defined for G_3 and so on. Note that the argument in Claim 3 implies that even the optimal algorithm must use at least $\lceil (\sum_{u \in N^i(v'_i)} r_u + r_{v'_i})/3 \rceil$ distinct colors for all v'_i. Thus, any coloring algorithm must use at least $\max_i \lceil (\sum_{u \in N^i(v'_i)} r_u + r_{v'_i})/3 \rceil$ distinct colors. The argument in Claim 1 and Claim 2 imply that the number of colors used by *B* is at most $\max_i \{\sum_{u \in N^i(v_i)} r_u + r_{v_i}\}$. Note that by design we have $\sum_{u \in N^i(v_i)} r_u + r_{v_i} \leq \sum_{u \in N^i(v'_i)} r_u + r_{v'_i}$ for all i. Thus, implying that *B* uses at most 3 times more color than any optimal coloring algorithm. ■

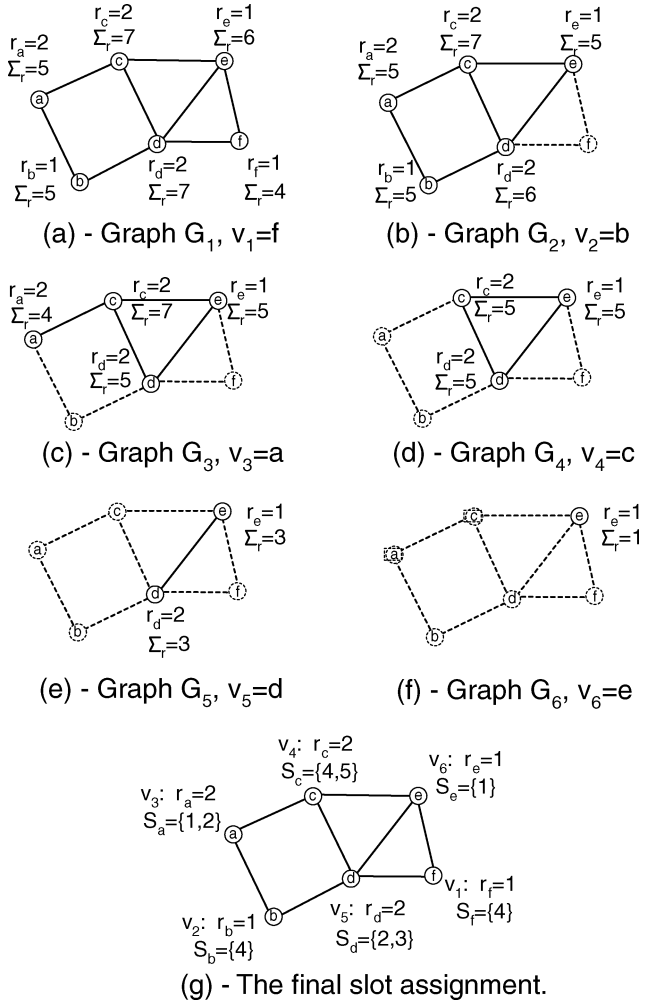

Fig. 9. An execution example of algorithm *B*.

C. The Fair Slot Assignment Scheme

Here we provide a formal definition of the slot assignment problem and show that it cannot be approximated. Then we propose to approximate it by using a bi-criteria approximation. As described earlier, the problem of providing fairness in our model is equivalent to assigning slots (colors) to access points in a superframe, such that the total number of slots (colors) assigned to an access point v in a superframe is proportional to the number of stations m_v associated with the access point. In addition, in our model fairness is provided with the additional goal of maximizing the throughput of the system. Recall that in our model the slot sizes are assumed to be at least Δ time units. Let $\tilde{D}$ denote the configured CFP in time units. Thus, the maximum number of slots in a CFP cannot exceed $R = \lfloor \tilde{D}/\Delta \rfloor$. Thus, the problem of providing fairness in our model can be formulated as the problem of finding the largest *min-slots-to-weight ratio*, ρ, such that there is a feasible superframe slot assignment in which each access point $v \in V$ is assigned at least $r_v = \lceil \rho \cdot m_v \rceil$

slots. Note that a superframe slot assignment is feasible if there is a coloring and frequency assignment of the underlying graph $G(V, E)$, in which node $v \in V$ has requirement r_v and the total number of colors used is at most $R = \lfloor \tilde{D}/\Delta \rfloor$. We now show that the slot-assignment problem, as stated above, is very hard not just to solve optimally but even to solve approximately. We then turn to a relaxation of the problem for which we develop efficient algorithms.

Claim 5: No polynomial time constant approximation algorithm is possible for the slot-assignment problem unless $P = NP$.

Proof: For contradiction, let us assume a c-approximation algorithm X for some constant c for the slot-assignment problem. This implies that algorithm X outputs a $\rho \geq \rho^*/c$ where ρ^* is the optimal *min-slots-to-weight ratio*. We show that X can be used to decide whether a unit disk graph with all unit requirements and a single frequency can be colored with three colors. As mentioned in Section VI-A, this is a NP-hard problem. Thus, implying that X does not exist unless $P = NP$. Given a unit disk graph $H(V, E)$. An instance of the slot-assignment problem is created with $G = H$, $m_v = 1$ for all $v \in V$, total number of slots (colors) $R = 3$ and a single frequency. Note that if H is 3-colorable then for G the value of $\rho = k \geq 1$ for some k and if H is not 3-colorable then $\rho = 0$. In the former case X outputs $\rho \geq 1/c$ while in the latter case X must output $\rho = 0$. Therefore, the return value of algorithm X can be used to determine if H is 3-colorable or not. This proves that the slot-assignment problem is hard to approximate. ■

In light of Claim 5, we turn to a different bi-criteria approximation for the slot-assignment problem. We define an (α, β)-approximation algorithm to be one that computes the value of ρ to within a factor α of the optimal *min-slots-to-weight ratio*, where the optimal is only allowed at most $\lfloor R/\beta \rfloor$ slots. Note that Claim 5 implies that a $(c, 1)$-approximation is not possible for any constant c. Note that allowing optimal to use less total number of slots is equivalent to restraining the optimal to use a larger slot size since the CFP size is fixed at $\tilde{D}$ time units.

We now present a (1, 3)-approximation algorithm X for the slot-assignment problem when there is a single frequency. The algorithm X works by guessing a value for the *min-slots-to-weight ratio* ρ. For a particular guess for ρ, it sets $r_v = \lceil \rho \cdot m_v \rceil$ for all $v \in V$. It then uses the unit-disk graph coloring algorithm *B*, presented in Section VI-B, to color the underlying graph. Let $f(\rho)$ denote the number of colors used by algorithm *B* for coloring this graph for the choice of ρ. Note that $f(\rho)$ is a monotonically non-decreasing function of ρ. Algorithm X uses a binary search over ρ to compute the largest value ρ' for which $f(\rho') \leq R$. Algorithm X then outputs $\rho = \rho'$. Since $f(\rho)$ depends on the r_v values, algorithm X needs only to consider those values of ρ, for which the quantity $\rho \cdot m_v$ is integral for some node v. Let ρ_H and ρ_L be the upper and lower bound of ρ' used by algorithm X, respectively. Thus, each time X sets either ρ_H or ρ_L, it lower or upper rounds the assigned value to the closest ρ such that $\rho \cdot m_v = \lceil \rho \cdot m_v \rceil$ for some node $v \in V$. In other words, for a considered set of r_v values that results from a guest ρ, if $f(\rho) > R$ then $\rho_H = \max_v \lfloor r_v/m_v \rfloor$. Otherwise, $\rho_L = \min_v \lceil r_v/m_v \rceil - \epsilon$, for a very small $\epsilon > 0$. The search ends when the gap between the two bounds is less than ϵ.

Claim 6: Algorithm X is a (1, 3)-approximation for the slot-assignment problem when only one frequency is used.

Proof: Let the optimal value of ρ for the slot assignment problem with $\lfloor R/3 \rfloor$ total slots be denoted by ρ_{OPT}. Thus, when the node requirements are set to $r_v = \lceil \rho_{\mathrm{OPT}} \cdot m_v \rceil$ for all $v \in V$ then the underlying graph G can be colored with at most $\lfloor R/3 \rfloor$ colors and for $\rho > \rho_{\mathrm{OPT}}$ the graph G requires strictly more than $\lfloor R/3 \rfloor$ colors. As shown in Section VI-B algorithm *B* is then able to color G for these node requirements with at most $3\lfloor R/3 \rfloor \leq R$ colors, implying that ρ' calculated by algorithm X satisfies $\rho' \geq \rho_{\mathrm{OPT}}$. ■

D. The Frequencies and Slot Assignment Algorithm

We now present an algorithm *C* for the joint coloring and frequency assignment problem. We will show that *C* is a 4-approximation algorithm. Algorithm *C* starts out by ordering the vertices of G in the same order $v_1, v_2, \ldots, v_n$ as ordered by algorithm *B* in Section VI-B and just like algorithm *B* assigns colors and frequencies to the vertices in the reverse order $v_n, v_{n-1}, \ldots, v_1$. Algorithm *C* uses a generalized First-Fit for its frequency and color assignment. Let $N_f^i(v_i) \subseteq N^i(v_i)$ be the set of neighbors of node v_i, in the graph G_i as defined in Section VI-B, that have been assigned frequency $f \in F$ by *C*. Note that the sets $N_f^i(v)$, $f \in F$ form a partition of $N^i(v)$ for all v in G_i. To determine the frequency assignment for node v_i, the algorithm *C* applies First-Fit to the set of colors assigned to nodes in $N_f^i(v_i)$ to determine the maximum color that would be assigned to v_i, assuming frequency f was assigned to node v_i by *C*. The algorithm *C* selects that frequency f for which this maximum color is minimized (ties broken arbitrarily). Let f be the frequency assigned by *C* to v_i then the set of colors assigned to v_i are those obtained by applying First-Fit to the set of colors assigned to nodes in $N_f^i(v_i)$.

```
Alg. C
For i = n down to 1
    minMaxColor = ∞
    f = 0
    For j = 1 up to |F| /* Compute best freq. f for v_i */
        C = r_{v_i}-th largest avail. color for v_i for freq. f_j
        If minMaxColor > C
            minMaxColor = C
            f = f_j
        End If
    End For
    Assign freq. f to v_i
    Assign r_{v_i} minimum avail. colors for freq. f to v_i
End For
```

Example 8: We consider again the graph G shown in Fig. 6. Note that algorithm *C* uses the same ordering of the vertices for G as the algorithm *B* in Section VI-B. This ordering is f, b, a, c, d, e. Let F consists of two frequencies f_1 and f_2. The frequency and slot assignment of algorithm *C* is depicted in Fig. 10. Algorithm *C* first assigns color 1 and frequency f_1 to node e. For the next node (node d) the smallest available colors for frequency f_1 are 2, 3 and for frequency f_2 are 1, 2. Hence, algorithm *C* assigns colors 1, 2 and frequency f_2 to node d. Note that at this point the set of available colors for node c are all colors except the color 1 of frequency f_1 and the colors 1 and 2 of frequency

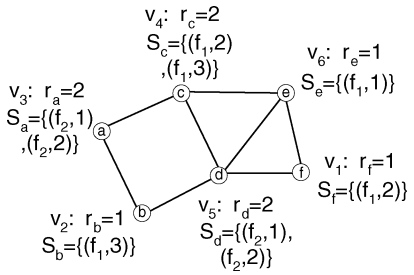

Fig. 10. An execution example of algorithm *C*.

f_2. Hence, *C* assigns colors 2 and 3 and frequency f_1 to node c. Next, algorithm *C* assigns colors to node a. Since its only neighbor with assigned colors is node c, algorithm *C* assigns colors 1 and 2 of frequency f_2 to node a. Node b is next and minimum available color is 3 for both frequencies f_1 and f_2. Hence, algorithm *C* assigns color 3 and frequency f_1 to node b. Finally, for node f the best frequency is f_1 and for this frequency algorithm *C* assigns it color 2. Thus, the total number of colors used by algorithm *C* for G with two frequencies is 3 which is optimal. ■

We now show that *C* is a 4-approximation algorithm for the unit disk coloring and frequency assignment problem.

Proposition 1: The maximum requirement $\max_v r_v$ is a lower bound on the number of colors used by any optimal solution.

Proof: This trivially follows from the fact that the colors assigned to any node must all be distinct. ■

Claim 7: The number of colors used by *C* is at most

$$\max_i \left\lceil \frac{\sum_{u \in N^i(v_i)} r_u + r_{v_i}}{|F|} \right\rceil + \max_v r_v.$$

Proof: For contradiction, let this not be the case and let the violation occur first time for node v_j. Then, for every frequency f, the maximum color among the set of colors obtained by applying First-Fit to node v_j by *C* is strictly greater than

$$\left\lceil \frac{\sum_{u \in N^j(v_j)} r_u + r_{v_j}}{|F|} \right\rceil + r_{v_j}.$$

Thus, for every frequency f the number of distinct colors assigned by *C* to nodes in $N_f^j(v_j)$ is strictly greater than $\lceil \sum_{u \in N^j(v_j)} r_u / |F| \rceil$, implying that

$$\sum_{u \in N_f^j(v_j)} r_u > \left\lceil \frac{\sum_{u \in N^j(v_j)} r_u}{|F|} \right\rceil$$

for all f. But this cannot happen since the sets $N_f^j(v_j)$, $f \in F$ form a partition of the set $N^j(v_j)$ and would thus imply that

$$\sum_{u \in N^j(v_j)} r_u = \sum_f \sum_{u \in N_f^j(v_j)} r_u > |F| \left\lceil \frac{\sum_{u \in N^j(v_j)} r_u}{|F|} \right\rceil \geq \sum_{u \in N^j(v_j)} r_u.$$

■

Claim 8: Any coloring and frequency assignment algorithm must use $\max_i \lceil (\sum_{u \in N^i(v_i)} r_u + r_{v_i})/(3|F|) \rceil$ colors.

Proof: Let $v_1', v_2', \ldots, v_n'$ be an ordered (multi)set of nodes as defined in the proof of Claim 4. Recall that v_i' is a node of G_i the center of whose unit disk has the minimum X-coordinate among all nodes in G_i. Thus, as shown in [20] the set of nodes $\{v_i'\} \cup N_i(v_i')$ does not contain an independent set of size more than 3 in G. Thus, the nodes $\{v_i'\} \cup N^i(v_i')$ require at least $\lceil (\sum_{u \in N^i(v_i')} r_u + r_{v_i'})/(3|F|) \rceil$ distinct colors in any feasible coloring of G, since the nodes assigned a particular color and a particular frequency must form an independent set of G. However, by construction we have $\sum_{u \in N^i(v_i)} r_u + r_{v_i} \leq \sum_{u \in N^i(v_i')} r_u + r_{v_i'}$. Thus, any coloring and frequency assignment algorithm must use at least $\lceil (\sum_{u \in N^i(v_i)} r_u + r_{v_i})/(3|F|) \rceil$ colors. Taking the maximum for all i we get the desired result. ■

Claim 9: *C* is a 4-approximation algorithm for the coloring and frequency assignment for unit disk graphs.

Proof: It can be shown that $\lceil 3x \rceil \leq 3\lceil x \rceil$ for all non-negative real numbers x. Thus, we have

$$\max_i \left\lceil \frac{\sum_{u \in N^i(v_i)} r_u + r_{v_i}}{|F|} \right\rceil \leq 3 \max_i \left\lceil \frac{\sum_{u \in N^i(v_i)} r_u + r_{v_i}}{3|F|} \right\rceil.$$

Recall that from Claim 7 the number of colors used by algorithm *C* is at most $\max_i \lceil (\sum_{u \in N^i(v_i)} r_u + r_{v_i})/|F| \rceil + \max_v r_v$, which is at most $3 \max_i \lceil (\sum_{u \in N^i(v_i)} r_u + r_{v_i})/(3|F|) \rceil + \max_v r_v$, which by Claim 8 and Proposition 1 is at most $3 + 1 = 4$ times the number of colors used by an optimal coloring and frequency allocation algorithm. ■

We now present an efficient slot assignment algorithm Y when there can be more than one frequency. The algorithm Y works exactly like algorithm X in Section VI-C, except that it uses algorithm *C* defined above, for the joint coloring and frequency assignment of the underlying graph, instead of the algorithm *B* presented in Section VI-B. Based on the notation introduced in Section VI-C it can be shown that Y is a (1, 4)-approximation algorithm for the slot-assignment problem.

VII. Finding the Interference Graph

This section presents a simple algorithm for estimating the interference graph when the AP locations are not known to the NOC. Our heuristic assumes that the transmission and the sensing ranges of an AP have bounded radii of R_T and R_S, respectively. In particular, it provides a good estimation of the interference graph in the case of circular region wireless channel, as described in Section III. In the following, we evaluate the quality of the estimated interference graph in this channel model. As defined in Section V, the *interference graph* $G(V, E)$ of the 802.11 network is defined by the set V of APs and a set of edges $(u, v) \in E$ between every pair of APs $u, v \in V$ which are at most $2 \cdot R_T + R_S$ apart. We denote this threshold distance by $\tilde{R} = 2 \cdot R_T + R_S$.

Note that the scheme for computing the interference graph has to be able to determine which APs are $\tilde{R}$ apart. In other words, for each AP v we want to compute the set of APs that are within a distance $2 \cdot R_T + R_S$ of v. One way this can be achieved is by having all the APs except v sense the channel for signal when v is transmitting. The APs close to v will be able to sense the transmission, while the ones far away won't sense

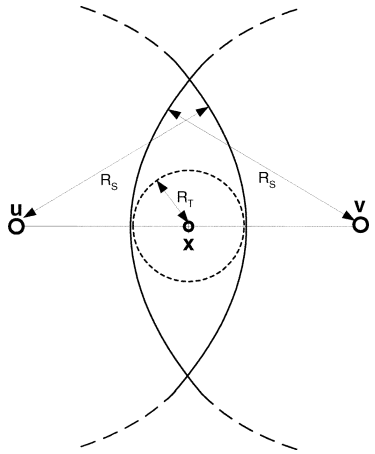

Fig. 11. The figure for proving the correctness of Claim 10.

any signal. By choosing the right transmission power for v it can be ensured that an AP senses the transmission of v if and only if it is within a distance of $\tilde{R}$ from v. However, this requires increasing the power of the AP v since at normal transmission power only APs which are at most R_S apart can sense transmissions. Moreover, the increase in the power can be as much as 4 times the normal transmission power of v which violate FCC limits. Thus, in the following we propose a different approach that does not require any increase in transmission power. The scheme works by silencing the network for a given time interval, as described in Section V, and allowing only one AP to transmit at a given time in the time interval. The APs take turn transmitting in the silent interval at their normal transmission power. The other APs attempt to sense the signal received and if they sense a signal they report it back to the NOC. Note that AP u is able to sense the transmission of AP v if and only if (u, v) is an edge in the sensing graph $H(V, A)$, which is defined as a graph of APs V in which there is an edge $(u, v) \in A$ between APs u and v if and only if the two APs are at most R_S apart. Thus, this scheme allows the NOC to construct the sensing graph H. Moreover, we assume that by evaluating the strength of the received signal, a receiving AP can estimate its distance from the transmitter. We use this estimated distance as the edge weight. We now show the following relationships between the sensing graph H and the interference graph G.

We define $H^i(V, A^i)$ to be the power graph $H(V, A)$ in which $(u, v) \in A^i$ if and only if u and v are at most i hops apart in H. For simplicity, we assume that every point in space is in the coverage area of some access points.

Claim 10: If $R_S/R_T \geq 4$ then $E \subseteq A^2$. In other words, the interference graph is contained in the square of the sensing graph.

Proof: Consider two APS $u, v \in V$ that are at most $\tilde{R} = R_S + 2 \cdot R_T$ apart and let x be the point at the center of the straight line between them, as shown in Fig. 11. Note that the distance from x to each one of the APs is at most $R_S/2 + R_T$. Let us denote this distance by d. Since $R_S \geq 4 \cdot R_T$, it follows that $d \leq R_S - R_T$. Thus, the circle centered at x with radius R_S is entirely included in the sensing ranges of both APs u and v. Since x is in the transmission range of some APs, there must be at least one AP inside this circle. Thus, the APs u and v are at most two hops apart in the sensing graph H. ■

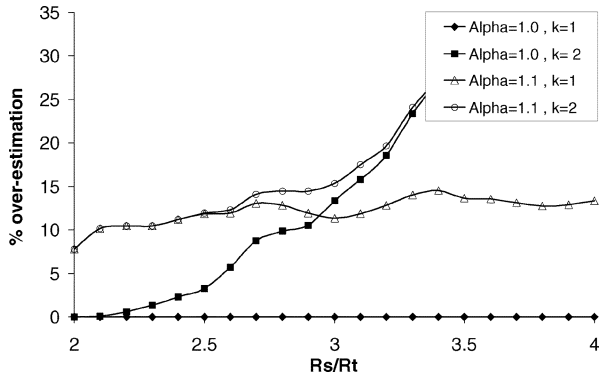

Fig. 12. Percentage of interference edges overestimated by our scheme.

By using similar geometric properties we are able to prove the following claim (proof omitted for lack of space).

Claim 11: If $R_S/R_T \geq 2.43$ then $E \subseteq A^3$. In other words, the interference graph is contained in the cube of the sensing graph.

Thus, given the sensing graph and the underlying physical layer model (the ratio of R_S and R_T) we can compute the interference graph quite accurately. Given the sensing graph H we use the following scheme to estimate the interference graph G. We use two parameters, k, the hop count and α, a scale factor. We add edge (u, v) to the interference graph if and only if either u and v are at most k hops apart in the sensing graph or the shortest distance between u and v in H is at most $\alpha\tilde{R}$.

Our simulations show that in practice our scheme for determining the interference graph works very well, and in fact it is sufficient to consider only the first and second power of H. In the following, we present our simulation results. For the simulation we used 80 APs uniformly distributed over a grid of size 1000×1000. The distribution of the AP's is picked to ensure complete coverage of the grid with R_T set to 100 units. We let R_S range from 200 to 400 units. We note that the results we observed are typical for many different AP placement schemes. For our simulations we used two values 1 and 2 for the hop count k and we used two values for α, namely 1.0 and 1.1. For each of these four possible choices for k and α we measure two parameters. An *overestimation* parameter that indicates the percentage of edges that are not in the interference graph and that our scheme selects. The other is a *violated length ratio* parameter that indicates the relative distance of the closest pair of APs $u, v \in V$ in the interference graph, $(u, v) \in E$, that our scheme has omitted. The violated length ratio of this edge is evaluated by taking the percentage ratio of $(\tilde{R} - d(u, v))/\tilde{R}$, where $d(u, v)$ is the geometric distance between the APs u and v. Both these results are for different values of the ratio R_S/R_T. Fig. 12 shows the results for the overestimation parameter while Fig. 13 shows the results for the underestimation parameter. These graphs show that the combination $k = 1$ and $\alpha = 1.1$ yields an accurate estimation of the interference graph for practical R_S/R_T ratios.

VIII. Simulation Results

We did extensive simulation to compare the performance of the MiFi system with that of the PCF and the DCF modes of 802.11 networks. We used three different performance metrics, described below. We quantify these performance metrics in

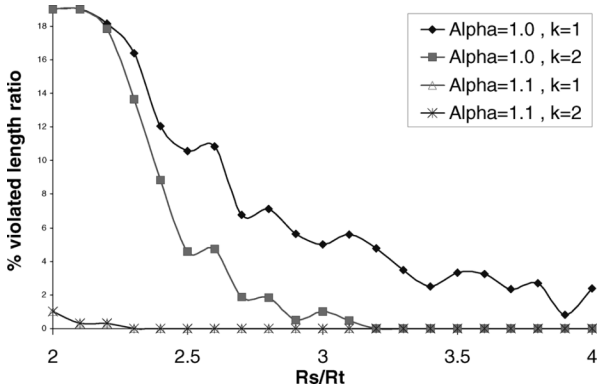

Fig. 13. Ratio of minimum length of edges not picked by our scheme to R_I.

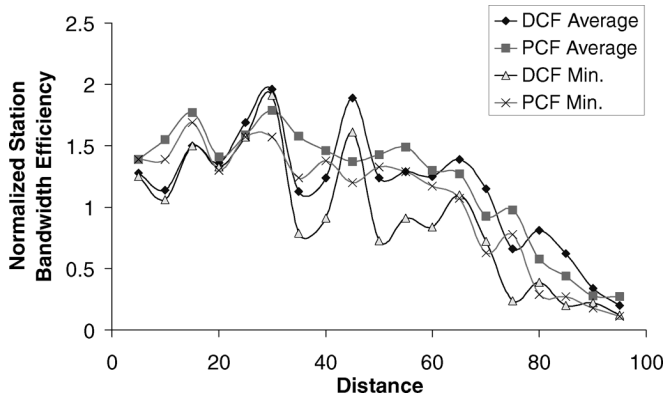

Fig. 14. The average and minimum normalized network efficiency in DCF and PCF modes.

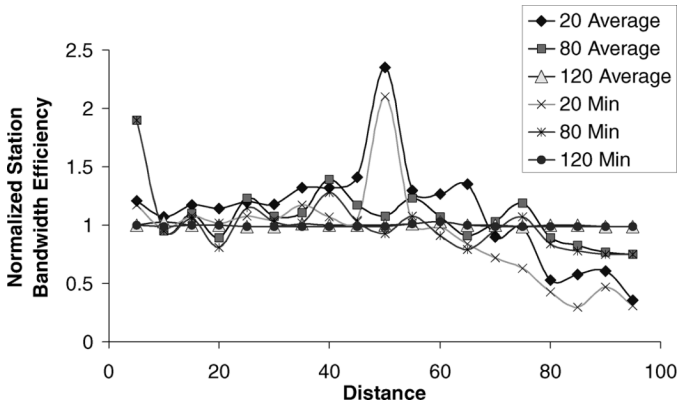

Fig. 15. The average and minimum normalized network efficiency of the MiFi system.

terms of the *efficient bandwidth* of the stations, which reflects the amount of data that the station transmits or receives successfully in a given time period. This parameter is a good measure of both the fairness and the throughput of the system. For lack of space we omit many of the details of our simulation setup and results. More detailed can be found in [1].

A. Simulation Setup

For the simulation we used a 802.11 network with 50 APs, uniformly distributed over a grid of size 1000×1000. Each AP has a transmission range of 100 units. The AP distribution is picked to ensure complete coverage of the grid and we follow the layout approach of [26]. We assumed 1000 mobile stations in the system that always have pending message to send and considered a message length of 1500 bytes (12000 bits) and, for a fair comparison, we assume that all the messages are transmitted with the same bit-rate. For both the PCF mode and the MiFi system we used a superframe size of 150 ms. Each simulation involved running the system for 1 minute and observing its behavior.

We tried our simulation both with a single frequency and with optimal frequency planning when there are three non-interfering frequencies, at different bit rates ranging from 1 to 10 Mbps per AP and for different CFP window sizes ranging from 10 ms to 140 ms. For lack of space the results presented, unless mentioned otherwise, are for a system with optimal frequency planning of three frequencies at 10 Mbps bit-rate, and CFP window set to 80 ms. We note that the results we observed are typical for all our simulations. We used an ideal network planning to illustrate the significant unfair behavior of the DCF and PCF mode even in optimally planned networks.

The first metric that we use to measure the relative performance of the MiFi system with respect to the 802.11 based WLAN is *intra-AP* fairness. Here we measure the *normalized* efficient bandwidth of the stations as a function of their distance from their associated APs. The normalized efficient bandwidth for a user is computed by dividing the efficient bandwidth of the user by the average efficient bandwidth of all the users that are associated with its AP. We measured both the average and the minimum values for the normalized efficient bandwidth for both the PCF and DCF modes of 802.11, and we compare it to that for the MiFi system. Note that an ideal (fair) system should have both a average and minimum normalized station efficient bandwidth of 1 at all distances from the AP. Fig. 14 shows our results for the 802.11 PCF and DCF mode for both the average and minimum normalized network efficiency of the stations within the transmission range of the AP. While Fig. 15 shows the MiFi system results for the same parameters for three different choices of the CFP window size: 20 ms, 80 ms, and 120 ms. Our results show that in the 802.11 case the normalized efficient bandwidth of the stations far away from the AP is close to 0, both for the average and the minimum metrics. On the other hand for the MiFi system both the average and the minimum normalized station efficient bandwidths are close to 1 at all distances even for the CFP window of 80 ms and the behavior of the MiFi system is very close to that of an ideal system even when the CFP window is of size 120 ms.

Our second metric is *inter-AP* fairness. Here we measure the minimum and average efficient bandwidth of *all* the stations in the system as a function of the size of the CFP window size (for PCF and MiFi) for both the MiFi system and the 802.11 DCF and PCF modes. The results are presented for the rate of 10 Mbps in Fig. 16. Our results indicate that the minimum value is almost 0, indicating starvation, for both PCF and DCF mode for all CFP window sizes. For the MiFi system however as the CFP window size is increased the minimum value rapidly approaches the average efficient bandwidth of *all* stations. Even for small CFP window sizes the gap between the minimum and the average is not significant for the MiFi system. These results

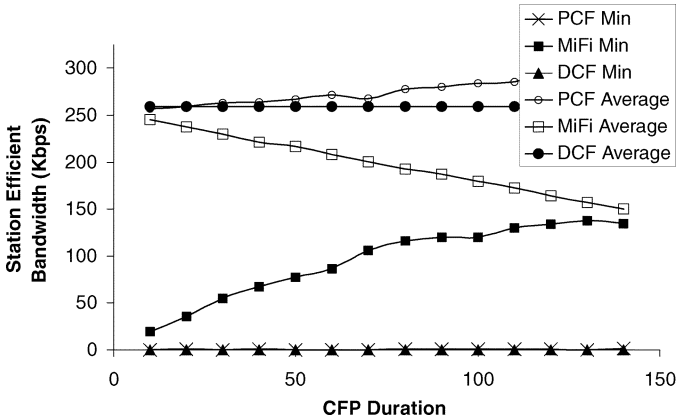

Fig. 16. The minimum and average efficient bandwidth of *all* the stations in date rate of 10 Mbps.

show that the MiFi system is starvation free and provides excellent fairness.

Our third metric is the overall system throughput. Note that the system throughput is the average efficient bandwidth of *all* stations times the number of stations. Thus, the results for the average efficient bandwidth in Fig. 16, when multiplied by 1000 gives the system throughput, as a function of the CFP window size, for both the MiFi system and the 802.11 DCF and PCF modes for bit rates of 10 Mbps. These results show that even for large CFP window size the overall system throughput of the MiFi system is comparable to that of the 802.11 networks. Moreover, we noticed that there is an optimal MiFi CFP size of 130 ms, at which the minimal efficient bandwidth of any station is maximized.

IX. Conclusion

In this paper, we introduced the MiFi system for ensuring fairness and supporting real-time services in large IEEE 802.11 networks. The scheme synchronizes the CFPs and CPs of all the APs and during CFPs it allows communication only at non-interfering cells. This guarantees that the system is free from the overlapping cell and the hidden node problems. Moreover, the scheme enables the system administrators to strike a balance between fairness and throughput.

References

[1] Y. Bejerano and R. Bhatia, "MiFi: A framework for fairness and QoS assurance in current IEEE 802.11 networks with multiple access points," in *Proc. IEEE INFOCOM'04*, 2004, pp. 1229–1240.

[2] P. S. Henry and L. Hui, "WiFi: What's next?," *IEEE Commun. Mag.*, vol. 40, no. 12, pp. 66–72, Dec. 2002.

[3] C. E. Koksal, H. Kassab, and H. Balakrishnan, "An analysis of short-term fairness in wireless media access protocols (poster)," presented at the ACM SIGMETRICS'00, Int. Conf. Measurement and Modeling of Computer Systems, Santa Clara, CA, Jun. 2000.

[4] T. Nandagopal, T.-E. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," in *Proc. MobiCom*, Aug. 2000, pp. 87–98.

[5] Q. Ni, L. Romdhani, T. Turletti, and I. Aad, "QoS issues and enhancements for IEEE 802.11 Wireless LAN," INRIA, Tech. Rep., 2002.

[6] D. Chalmers and M. Sloman, "A survey of quality of service in mobile computing environments," *IEEE Commun. Surveys*, vol. 2, no. 2, 1999.

[7] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN," in *Proc. MobiCom*, 2000, pp. 167–178.

[8] T. Ozugur, M. Naghshineh, P. Kermani, C. M. Olsen, B. Rezvani, and J. A. Copeland, "Balanced media access methods for wireless networks," in *Proc. MobiCom*, 1998, pp. 21–32.

[9] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," in *Proc. IEEE INFOCOM*, 2001, pp. 209–218.

[10] D. Qiao and K. G. Shin, "Achieving efficient channel utilization and weighted fairness for data communications in IEEE 802.11 WLAN under the DCF," in *Proc. IWQoS*, 2002, pp. 227–236.

[11] A. Veres, A. T. Campbell, M. Barry, and L.-H. Sun, "Supporting service differentiation in wireless packet networks using distributed control," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 2081–2093, Oct. 2001.

[12] S.-T. Sheu and T.-F. Sheu, " DBASE: a distributed bandwidth allocation/sharing/extension protocol for multimedia over IEEE 802.11 ad hoc wireless LAN," in *Proc. IEEE INFOCOM*, 2001, pp. 1558–1567.

[13] A. Kopsel and A. Wolisz, "Voice transmission in an IEEE 802.11 WLAN based access network," in *Proc. Workshop on Wireless Mobile Multimedia (WoWMoM)*, 2001, pp. 23–32.

[14] C. Coutras, S. Gupta, and N. B. Shroff, "Scheduling of real-time traffic in IEEE 802.11 wireless LANs," *Wireless Netw.*, vol. 6, no. 6, pp. 457–466, 2000.

[15] M. Veeraraghavan, N. Cocker, and T. Moors, "Support of voice services in IEEE 802.11 wireless LANs," in *Proc. IEEE INFOCOM*, 2001, pp. 488–497.

[16] S. Mangold, "Coexistence of overlapping basic service sets," in *Proc. Mobile Venue'02*, 2002, pp. 131–135.

[17] S. Mangold, L. Berlemann, and G. Hiertz, "QoS support as utility for coexisting wireless LANs," presented at the Int. Workshop on IP Based Cellular Networks (IPCN), Paris, France, 2002.

[18] B. O'Hara and A. Petrick, *The IEEE 802.11 Handbook: A Designer's Companion*, 2nd ed. New York: IEEE Press, 2005.

[19] T. S. Rappaport, *Wireless Communication Principle and Practice*. Englewood Cliffs, NJ: Prentice Hall, 1996.

[20] M. V. Marathe, H. Breu, H. B. Hunt, III, S. S. Ravi, and D. J. Rosenkrantz, "Simple heuristics for unit disk graphs," *Networks*, vol. 25, pp. 59–68, 1995.

[21] D. S. Hochbaum and W. Maass, "Approximation schemes for covering and packing problems in image processing and VLSI," *J. ACM*, vol. 31, pp. 130–136, 1985.

[22] U. Feige and J. Kilian, "Zero knowledge and the chromatic number," *Comput. Syst. Sci.*, vol. 57, pp. 187–199, 1998.

[23] B. N. Clark, C. J. Colburn, and D. S. Johnson, "Unit disk graphs," *Discrete Math.*, vol. 86, no. 1-3, pp. 165–177, 1990.

[24] M. Stumpf, A. Graf, and G. Weibenfels, "On coloring unit disk graphs," *Algorithmica*, vol. 20, no. 3, pp. 277–293, 1998.

[25] R. Peeters, "On coloring j-unit sphere graphs," Dept. Economics, Tilburg Univ., Tech. Rep., 1991.

[26] A. Hills, "Large-scale wireless LAN design," *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 98–107, 2001.

Yigal Bejerano received the B.Sc. degree (*summa cum laude*) in computer engineering in 1991, the M.Sc. degree in computer science in 1995, and the Ph.D. degree in electrical engineering in 2000, from the Technion–Israel Institute of Technology, Haifa, Israel.

He is currently a Member of the Technical Staff (MTS) at Bell Laboratories, Lucent Technologies, Murray Hill, NJ. His research interests are mainly management aspects of high-speed and wireless networks, including the areas of mobility management, network monitoring, topology discovery, quality of service (QoS) routing, wireless LAN, and wireless mesh networks.

Dr. Bejerano is on the technical program committee of the IEEE INFOCOM 2002–2006 conferences.

Randeep S. Bhatia received the B.Tech. degree in computer science and engineering from the Indian Institute of Technology, Delhi, the M.S. degree in mathematics and computer science from the University of Illinois at Chicago, and the Ph.D. degree in computer science from the University of Maryland, College Park.

He is currently with Bell Labs, Lucent Technologies, Murray Hill, NJ, working on network design, traffic engineering, and scheduling algorithms. His current research interests are in the area of QoS for emerging multimedia services in next-generation data networks.