Adaptive Diversity Based Spectrum Allocation in Single-Radio Wireless Ad Hoc Networks

Marzieh Veyseh
marzieh@soe.ucsc.edu
Computer Engineering Department,
University of California, Santa Cruz
Santa Cruz, CA 95064, USA

J.J. Garcia-Luna-Aceves
jj@soe.ucsc.edu
Computer Engineering Department
University of California, Santa Cruz
Santa Cruz, CA 95064, USA

Hamid R. Sadjadpour
hamid@soe.ucsc.edu
Electrical Engineering Department,
University of California, Santa Cruz
Santa Cruz, CA 95064, USA

***Abstract*—A new cross-layer design taking advantage of OFDMA in ad hoc networks is presented. OFDMA technology is exploited at the physical layer to improve data rate through multiuser diversity and to enhance channel throughput by enabling multiple concurrent transmissions over orthogonal subchannels, each consisting of a group of tones or subcarriers. The proposed Subchannel Selection Algorithm (SSA) addresses the distribution of subchannels and the new Tone Assignment Algorithm (TAS) takes advantage of fading and is adapted to the limitations of ad hoc networks. TAS operates alongside the signaling of the resulting medium access control (MAC) protocol called Concurrent Communication medium Access or CoCo-MAC. The new MAC addresses the synchronization requirements of OFDMA and the needs of the tone assignment algorithm, and also enables concurrent initiation of data transmissions from multiple nodes to the same receiver or from a single transmitter to multiple receivers. We present analysis and simulation results on the throughput advantages of our technique compared to previous spectrum allocation and MAC protocols based on the avoidance of multiple access interference.**[1]

I. Introduction

Attaining high channel throughput is a major goal in the design of medium access control (MAC) and physical (PHY) layer schemes. The objective is to enhance throughput by either (a) enabling *concurrency*, (b) adding *diversity*, or providing (c) *adaptivity* in the allocation of resources.

To achieve concurrency, previous MAC protocols utilize orthogonal multichannel networks [1], [2], CDMA [3], MIMO [4] (also provides spatial diversity), and network coding.

Many multichannel FDMA MAC protocols have been proposed in the past [1], [2], where the entire spectrum is divided into orthogonal channels, and nodes switch between such channels to enable concurrent data transmissions. The drawbacks of these techniques are channel switching delays, restrictions on the number of available orthogonal channels, and the inability to deploy dynamic bandwidth allocation techniques. CDMA-based MAC protocols enable concurrent transmission of data over a wider spectrum by multiplying the transmitted signal with a unique code specified for that transmission. However, the drawback of this approach is the need for complex equalization techniques and inability to transmit more than one packet at a time.

Recent results have demonstrated that the capacity of wireless ad hoc networks can significantly improve as a result of spatial diversity if nodes are endowed with multiple interfaces/radios in the presence of multiple non-overlapping channels [5]. However, it is not realistic to utilize as many radios as the number of non-overlapping channels that may be available in a network. Channel assignment and medium access is an even more challenging problem in such networks [4], [6]. TDMA has also been considered for adaptive adjustment of time slots' length among interfering neighbors, however, there is no concurrency of transmissions around receivers, while the adaptive allocation of time is not efficient.

Orthogonal Frequency Division Multiple Access (OFDMA) has been selected for use in multi-user environments (e.g., IEEE 802.16 [7] and DVB [8]) employing OFDM technology due to its ability to combat the multipath effects of wireless channels, and to facilitate the concurrency of transmissions. In OFDM systems, *subcarriers* or *tones* are orthogonal carriers of lower-rate input data streams that mitigate multipath effects. In OFDMA, a group of non-overlapping tones called a *subchannel* can be assigned to each user, thus enabling simultaneous data transmission while intelligent assignment of subchannels based on wireless channel fading results in *multiuser channel diversity*.

Previous work focusing on channel assignment for OFDMA infrastructure-based networks [9]–[12] focuses on fast heuristics for centralized scheduling, which is not applicable to ad hoc networks. The adoption of OFDMA in ad hoc networks has been explored by a few recent works [13]–[15]. These schemes focus on resource allocation algorithms in a multi-antenna environment, or routing, and do not provide a MAC protocol to attain multiuser diversity using single-radio nodes. Scheduling in time and frequency for mesh networks in which routers are responsible for channel assignment is discussed in [16]. On the other hand, the work reported in [17] focuses on a prototype multiuser dynamic OFDMA in a realtime WLAN testbed and does not address ad hoc networks.

To our knowledge, no previous MAC protocol has been designed for ad hoc networks using an OFDMA physical layer to achieve multiuser channel diversity while exploiting concurrency.

Section II of this paper presents an overview of OFDMA networks and the synchronization restrictions of OFDMA ad hoc networks.

Sections III, IV, and VI present the new diversity-based adaptive spectrum allocation approach for OFDMA ad hoc networks. Our approach groups tones into subchannels and assign to each node multiple non-overlapping subchannels unique to the subchannels of nodes within its one-hop and two-hop neighborhood to avoid multi-access interference (MAI). Our approach enables multiuser diversity in the management of tones within a subchan-

[1] This research was partially sponsored by the U.S Army Research Office under grant number W911NF-05-1-0246, by the National Science Foundation under grant CCF-0729230, and by the Baskin Chair of Computer Engineering. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies of the U.S. Government.

nel. A tone assignment algorithm is introduced for each node to use for the tones within the assigned subchannels based on fading to achieve multiuser diversity and fair resource allocation and eventually increase channel throughput. We describe the MAC signaling that addresses synchronization restrictions of OFDMA and integrates the required physical-layer signaling with the MAC layer control messages to achieve a cross-layer solution. Section VIII presents analysis and simulation results that show the channel throughput advantages of our technique compared to traditional MAC protocols based on contention-based avoidance of MAI.

II. OFDMA Overview

In OFDM, the input data stream is split into a number of lower-rate parallel orthogonal *Tones* and is transmitted using a single carrier frequency. Basically, OFDMA is similar to OFDM technology however designed specifically to be used in a multi-user environment. The idea is to group multiple tones into a subchannel and each user transmits data on the assigned subchannel while sending no information over the rest of the tones. Therefore, all users send data at the same time on different parts of the spectrum. In addition to the concurrency attained, an advantage of OFDMA is that each user can be assigned the best tone from a selection of possible tones. Hence, each user experiences better channel condition and takes advantage of fading. This is based on the fact that the probability of facing a deep fade by all users on a specific tone is negligible. Thus *multiuser channel diversity* gain is attained. Meanwhile, a node can utilize multiple subchannels at the same time for communication. So the number of utilized subchannels can change based on the network demands. We will observe, how this adaptive allocation of bandwidth improves performance.

To be able to adopt the same OFDMA concept to ad hoc networks in which a multi-transmitter scenario is possible, tone orthogonality must be maintained at all receivers. In this case, transmitters should use non-overlapping parts of the bandwidth to send their data, however, because packets are sent using the same carrier frequency, the received signal at a receiver is the addition of all OFDM symbols transmitted over the air. For the receiver to be able to decode any of the transmissions successfully, a *quasi-synchronous* network is required [18], [19], meaning that all transmitters must start transmitting data at the same time. In this case, the time offsets among received signals is limited to the propagation delay and can be incorporated as part of the channel impulse response. Thus, the offset can be compensated as part of the channel equalization performed at the receiver if the added cyclic prefix to each frame is longer than the channel delay spread plus the relative propagation delay among users. Given that in practice the cyclic prefix is designed to be very long, and the propagation delays between nodes are relatively short in a typical ad hoc wireless network, this assumption is not restrictive [19].

In this work, we assume that the time for data transmission is divided into time slots, and we address the required signaling to create a *quasi-synchronous* network in Section VI.

III. Cross-Layer Solution for OFDMA Adoption in Ad Hoc Networks

Tones are grouped into non-overlapping subchannels. To avoid MAI, each node is assigned multiple subchannels that do not overlap to the subchannels assigned to one-hop and two-hop neighbors. The subchannels assigned to each node work together as one *channel*. As shown in Figure 1, a common transmitter (C-Tx) uses the tones within the assigned channel to concurrently transmit data to multiple receivers. A common receiver (C-Rx) uses the tones within the assigned channel to concurrently receive data from multiple transmitters. If one node acts as a C-Tx and the other as a C-Rx at the same time, there will be interference at the C-Rx; therefore, the channel should be non-overlapping to any one-hop neighbor's channel. The channel should also be unique to any two-hop neighbor's channel, in case they both act as a C-Tx or both act as a C-Rx. The proposed MAC signaling ensures quasi-synchronous operation to allow this. There is no need for three hop away uniqueness, because as shown in Figure 1, if there is a node three hops away that is currently acting as a C-Tx, there will be a conflict of timing between the two-hop away transmitters. Therefore, with CoCo-MAC, the C-Rx does not establish a transmission from node i at the same time as the C-Tx.

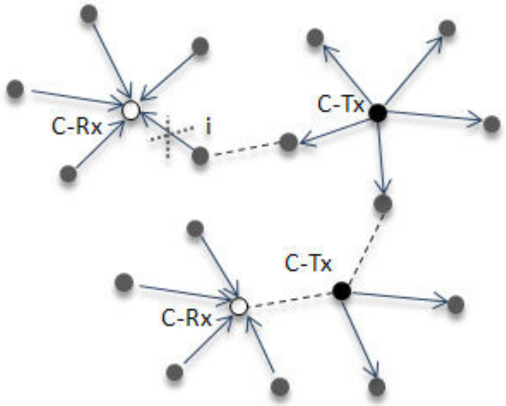


Fig. 1. Illustration of the subchannel selection protocol

In this design we assume that each subchannel includes the minimum number of tones possible, and as a result, the maximum number of subchannels are available (802.16, up to 96 subchannels [20]). In this work, we address the issue of subchannel assignment to avoid MAI by introducing SSA, and propose a simplified tone assignment algorithm within the assigned subchannels to exploit multiuser channel diversity. We will address the synchronization restrictions of OFDMA as well as the requirement of channel state information at the transmitters and receivers with our MAC signaling.

IV. Subchannel Selection Algorithm (SSA)

Our proposal entails two stages of sub-channel (Schannel) assignment to reduce overhead and number of message exchanges necessary to allocate non-interfering Schannels to intrusive neighbors: a hashing stage and a sequential-selection stage. The assignment needs to run only when topology changes occur in the neighborhood. The first stage uses a common hashing function to assign Schannels to interfering nodes, while the second stage attempts to assign the Schannels that remain unassigned due to the randomness of the first stage. For traffic pattern, we assume that each node wants to communicate with K of its randomly selected one-hop neighbors.

A. Hashing Stage

With OFDMA, a node can utilize n Schannels concurrently for its transmission to another node [21] with a BW n times the BW on each Schannel. With SSA, the smaller the number of interfering nodes in the network, the larger the number of Schannels that are assigned to the node, which results in adaptive bandwidth selection. Each node needs to create a conflict graph $G(U,E)$ based on the interfering neighborhood information it receives from its one-hop neighbors. The idea is to allocate Schannels to nodes based on their traffic requirements; therefore, each node u is represented by K vertices in the conflict graph, where u wants to communicate with K of its randomly selected direct neighbors. Each vertex, u^k, is connected to all of its interfering vertices. The interfering vertices for u^k represent the nodes up to H hop away from u, as H is the number of hops that the interference traverses in the network (we assume $H = 2$, although it could be different). In addition, u^k is connected to all u^j $j \neq k$, because vertices presenting a common node should obtain different Schannels. Each vertex u^k owns a traffic ID, TID_u^k. At this stage each vertex, u^k, is calculating I_u Schannels for all vertices within the conflict graph based on a known hashing function, where $I_u = \left\lfloor \frac{N^S}{(f_u+1)} \right\rfloor$ and N^S is the number of available Schannels and f_u is the number of conflicting vertices for node u. An array of common seeds among all vertices is used to generate multiple Schannels for each vertex. To calculate the *i*th Schannel for vertex u^k:

$$Schannel_u^k(i) = f_{hash}(TID_u^k \oplus seed(i)) \quad for\ i = 1,..,I_u$$

After the hashing phase, a vertex compares its selected Schannels with the Schannels of all interferes. Note that this is done locally, because the hash function, TID_u^k, seeds, and I_u for all $u,k \in G(U,E)$ are known at each vertex. If a selected Schannel for a vertex is not the same as $Schannel_u^k(i)$ for all $u,k,i \in G(U,E)$, that Schannel is added to a Confirmed List ($ConList$). Then each node advertises $ConList_u^k$. At this point, each node creates a new list of available Schannels for its vertices which excludes $ConList_u^k$ belonging to its interferes. This stage attempts to randomly select multiple Schannels for nodes while the next stage will solve the assignment issue sequentially for the remaining Schannels.

B. Sequential Selection Stage (SEQ)

The hashing stage attempts to give each node, u, up to $I_u \times K$ unique Schannels randomly. However, a variable number of Schannels can be given to a vertex, depending on the conflict graph of the vertex's neighbors. Therefore, we propose a sequential technique.

1) Assumptions and Notations: At this stage, each node creates a new conflict graph, $G(U,E)$, where the node is represented by vertex u. Vertex u is connected to vertices representing the nodes up to H hops away from u as interfering neighbors. An *Schannel priority* $CP_u(c)$ for each remaining Schannel c of vertex u is assigned.

An Schannel priority $CP_u(c)$ may have two possible values: *low* (*l*) and *high* (*h*). An Schannel with priority h on vertex u must be assigned a priority l on all interfering vertices. A priority value of *high* denotes a low collision possibility, and a priority value of *low* means a high collision possibility for the Schannel being utilized by the node. All Schannels chosen for a node by the hashing stage receive priority h prior to this stage. Also, all Schannels chosen for a node's interferes receive priority l on this node.

The success rate of Schannel c on vertex u is denoted by $ps_u(c)$. SEQ aims to achieve the maximum success rate, P_u over all Schannels on each node, while minimizing the standard deviation of P_u, denoted by $\rho = \sigma_u(P_u)$, over all nodes. To maximize P_u, SEQ assigns priority of h to as many Schannels of node u as possible. SEQ minimizes ρ by allocating priorities during the assignment in such a way that nodes receive equal opportunity of success according to their interferes.

The priority table, CPS_u, contains $CP_u(c)$ for all of the Schannels of node u, and $CPS_{N(u)}$ is the priority table for $N(u)$, i.e., the set of direct neighbors of vertex u. At the start of this stage, $CP_u(c)$ for all c, except the ones confirmed by the hashing stage move to the priority of *start* (*s*), which has $ps(c) = 0$, and during the algorithm's operation changes to either h or l. SEQ requires the exchange of priority tables among one-hop neighbors at the end of each round (*CPS update*). Each message contains information for each one-hop neighbor. Therefore, each node acquires priority information for its two-hop neighbors. To carry this information, a designated *control* Schannel is used.

2) Priority assignment: SEQ for each Schannel c assigns priorities by finding the highest priority vertex in the conflict graph, which we call *robust*. To assure fairness, the highest-priority vertices are those with the lowest probability of success P_u where:

$$P_u = \sum_{c=1}^{N^S} ps_u(c)/N^S$$

and $ps_u(c) = 0.9$ if $CP_u(c) = h$, and $ps_u(c) = 0.01$ if $CP_u(c) = l$. At each round, after a vertex detects itself to be *robust*, it calculates a probability of success for the first Schannel with priority s:

$$p_u(c) = 1 - \frac{N_u^l \times l + 100 \times N_u^h \times h}{N_u^l + 100 \times N_u^h} \tag{1}$$

where N_u^l is the number of vertices with $CP(c) = l$ and N_u^h is the number of vertices with $CP(c) = h$ in $CPS_{N(u)}$. We set $l = 0.01$ and $h = 0.9$ so that if $p_u(c) > 0.9$, then $CP_u(c) \leftarrow h$; otherwise, $CP_u(c) \leftarrow l$.

If $CP_u(c)$ of the current *robust* vertex is computed to be l, there already exists a $CP_v(c) = h$ on the interfering vertex v, and u can safely update its Schannel priority. However, if the calculated $CP_u(c)$ equals h, the vertex can only make the change if it is the sole winner among all *robust* vertices, i.e., has the maximum PID when $PID(x) = f_{hashPID}(ID(x) \otimes c)$ while x represents all *robust* vertices with $P(x) = min_u(P_u)$. The winner among all *robust* vertices is $robust_{winner} = v$ when $PID(v) = max_x(PID(x))$. The vertices that do not make any updates at this round are called *torpid*.

Lemma 4.1: After SEQ converges, only a single vertex u gets $CP_u(c) = h$ on Schannel c, among its interfering vertices.

Proof: Based on Eq. 1, if $N_u^h \geq 1$ (i.e., at least one vertex with $CP(c) = h$ in $CPS_{N(u)}$), the resulting $p_u(c) < 0.9$ and the assigned priority to $CP_u(c)$ must be l. ■

Lemma 4.2: Within a finite time after all topology changes has stopped, SEQ converges, and no more *CPS updates* are transmitted.

Proof: For each Schannel, only one vertex u within the conflict graph updates $CP_u(c)$ to h as a *robust* vertex. After sending a *CPS update* to all its neighbors, the recipients of this

```
Algorithm : Tone Assignment (TAS)

Input: Communicating-neighbor-list
Output:Tone-list(Communicating-neighbor-list),
modulation(Communicating-neighbor-list)

for $t = 1 : N^t, (N^t = number\ of\ tones)$
    $find\ M^j(t) \leftarrow max_j(M^{i,j}(t)),$
    $(j \in communicating - neighbor - list)$
for $m = 2 : M^j(t)\ (m \in \{2,\ 16,\ 64,\ 256\})$
    $rate^m(j) = rate^m(j) + m$
    $add\ t\ to\ Tone - list(j)$
if $rate^m(j) \geq R_j$
    $eliminate\ j\ from\ Communicating - neighbor - list$
    $modulation(j) = m$
if $t < N^t\ and\ Communicating - neighbor - list == empty$
    $R_j = \infty\ for\ all\ j$
    $run\ TAS\ for\ (N^t - t)$
```

Fig. 2. Pseudo-code description of TAS

update were *torpid* in the previous round. Because the priority selection of *torpid* vertices is a result of the selection of *robust* vertices, their *CPS updates* do not change the priority of the *robust* vertices. The *robust* vertices that assign $CP(c) = l$ must have been *torpid* in the previous round so again their updates do not affect the *robust* vertices selection. Therefore, all vertices change from s to either l or h within a finite time, and no more updates are sent. ■

Lemma 4.3: After SEQ converges, P_u on each vertex is maximized while standard deviation $\rho = \sigma_u(P_u)$ across vertices is minimized.

Proof: When $CP_u(c)$ on vertex u is assigned h, it means that it had the lowest P_u. Therefore, at no point in time a change can increase the vertex's P_u from others, unless the other links have an equal P_u; hence, the standard deviation is minimized while $\sum_{c=1}^{N^S} ps_u(c)/N^S$ is maximized. ■

In an OFDMA-based network, many Schannels could be available for use based on the subcarrier or tone grouping. If the average number of one-hop neighbors of nodes is the network degree *Ne*, the average number of Schannels required to assign at least one Schannel to each node is given by $Ne(Ne-1)+1+Ne = Ne^2 + 1$. Table I shows the number of Schannels needed for different network degrees when typical number of Schannels for OFDMA network can go up to 96 [20].

TABLE I
NETWORK DEGREE AND THE REQUIRED NUMBER OF SCHANNELS FOR IDEAL ASSIGNMENT

Network Degree	3	4	5	6	7	8
Required number of Schannels	10	17	26	37	50	65

Theorem 4.4: After a finite number of message exchanges, SEQ fairly assigns valid Schannel priorities to all Schannels on all nodes in such a way that $\sigma_u(P_u)$ is minimized.

Proof: The theorem follows directly from Lemmas 4.1, 4.2, and 4.3. ■

V. TONE ASSIGNMENT ALGORITHM (TAS)

A. Tone Assignment Algorithm (TAS)

The key feature of our solution is to exploit fading in such a way that multiuser diversity is achieved. To reduce complexity, we assume that the transmission power is selected equally for all tones. Also a common transmitter or a common receiver is aware of the rate requirement for neighbors. Assume that a transmission is about to be set between node i and node j. Assuming slow fading, if node i is responsible for choosing the best tones for transmission to node j, channel state information, $H^{i,j}$ and power of interference on all tones should be known at i. However, only node j can estimate the channel after receiving pilot signals from node i, and transfer that information back to node i. Node i should be aware of the sensitivity parameters of node j's radio to calculate number of bits decodable at node j on all tones. While this method is correct, it entails the transfer of many informational bits back to the transmitter, as well as computational complexity. Accordingly, we propose to have node j calculate the number of decodable bits based on its own radio sensitivity parameter, after estimating $SINR^{i,j}(k)$ on tone k by receiving node i's pilot signals. With $SINR^{i,j}(k)$ known at node j, it can calculate how many bits can be received on each tone based on the minimum SINR required for a modulation (QPSK, 16QAM, 64QAM, 256QAM) to be decodable at the receiver. This maximum attainable rate is called $M^{i,j}(k)$. Now, if node i is a common transmitter, it only would need to be aware of the value of $M^{i,j}(k)$ to be able to take advantage of diversity using our algorithm. On the other hand, if node j is a common receiver, it has to calculate M for all of the transmitters and run our algorithm to assign tones. We will discuss subsequently how the MAC signaling transfers this information between receivers and transmitters. In this section we explain how $M^{i,j}(k)$ is used to assign tones among multiple neighbors.

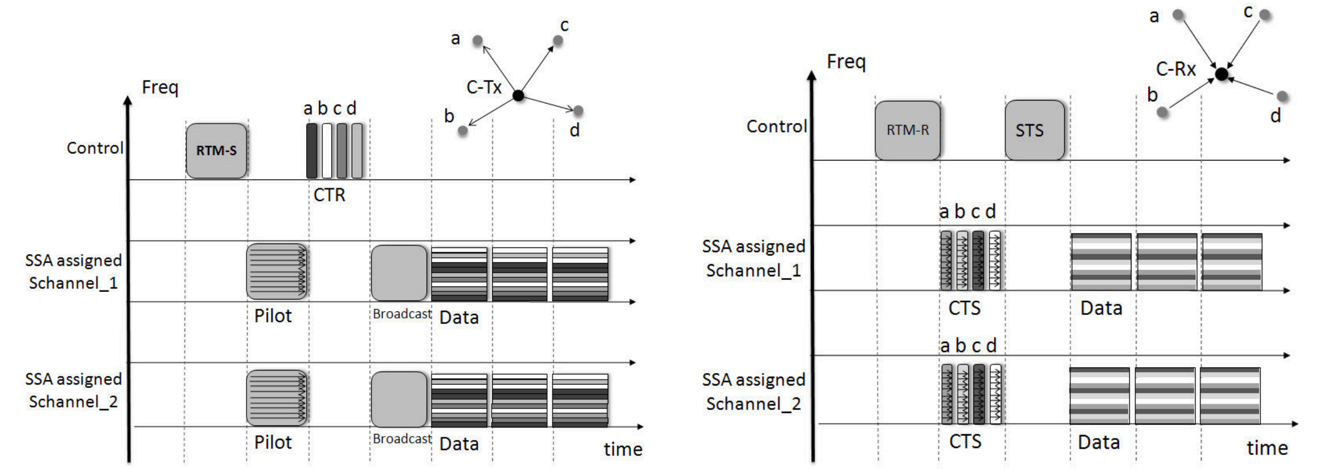
TAS runs through tones and for each tone, k, finds the neighbor with the maximum value of M (maximum number of bits decodable at the receiver) and allocates that tone to that neighbor:

$$choose\ neighbor\ n = \arg\max_{n \in K}(M^{i,n}(k))$$

where K is the number of neighbors that this node is using for communication. A table is created for each neighbor with the sum of number of bits on all of the assigned tones for each modulation. Note that all modulations with bit rate smaller that the maximum decodable rate could be used on a tone. Therefore, for a neighbor, multiple sets of accumulated attainable rates are calculated. Since we assume that all tones assigned to a neighbor should carry the same number of bits (same modulation), the accumulated rate is different for each modulation. During the run, if any of the rates for any of the modulation meets the rate requirements for a neighbor, TAS selects that modulation for that neighbor, eliminates the neighbor from the rest of the assignment, and move on to the next tone. If all nodes, reach the required rate and still some tones are left, the algorithm will start over and distributes those tones among neighbors and selects the best modulation (highest attainable rate) for each neighbor.

VI. COCO-MAC

The MAC protocol is responsible for (a) allowing the estimation of the channel state at receivers by transmission of pilot signals from transmitters to receivers prior to data transmission;



(a) Illustration of MAC signaling when a common transmitter sends to multiple receivers

(b) Illustration of MAC signaling when a common receiver receives from multiple receivers

Fig. 3. Illustration of MAC signaling

(b) exchanging information regarding the number of bits that can be received at neighbors on each tone in the common transmitter scenario; (c) exchanging the information regarding the allocated tones from a common transmitter or common receiver to the corresponding neighbors; and (c) performing synchronization to avoid loss of orthogonality in OFDMA in a multi-transmitter situation.

1) Common Transmitter (C-Tx):

1) C-Tx transmits an RTM-S (Request to Multiple Send) over the control Schannel. The RTM-S contains clock frequency reference, subjected receivers' address and time reference, as well as schedule for transmitting a CTR (Clear to Receive).
2) C-Tx sends pilot data over the subchannels that are assigned via SSA.
3) Nodes that successfully receive the RTM-S message estimate the fading for all tones within the allocated subchannels. Then, based on the SINR and the sensitivity of the receiver, they calculate the maximum number of bits that can be received on each tone.
4) The receivers transmit a CTR (Clear To Receive) message following the schedule sent by the C-Tx to avoid interference at the sender. The CTR message contains a table indicating the calculated $M(k)$ for each tone and the time reference information originally sent by the C-Tx.
5) All nodes neighboring the receivers are aware of the clock reference time of the C-Tx and can fix their time to avoid loss of orthogonality in a multi-transmitter scenario.
6) C-Tx runs the tone assignment algorithm and assigns a grouping of tones to each neighbor.
7) The first OFDM frame is a broadcast frame and contains the list of the assigned tones for each neighbor.
8) C-Tx starts sending data to multiple nodes using the allocated tones.
9) The receivers obtain an OFDM frame that contains a null value on some tones. Each node decodes the entire frame and after the FFT module filters the assigned tones, it detects its own data.

Figure 3(a) illustrates an example for this scenario when a C-Tx is attempting to transmit data to nodes a, b, c and d. After transmission of RTM-S, all neighboring nodes can adjust their clock frequency and their time reference to the common transmitter and create a *quasi-synchronous* scenario as explained in Section II. The pilot signals sent by the C-Tx are known signals that occupy most of the tones on the pre-assigned subchannel to enable estimation of channel at the receiver side. After $M^{C-Tx,a}(k)$, $M^{C-Tx,b}(k)$, $M^{C-Tx,c}(k)$, $M^{C-Tx,d}(k)$ is calculated at a, b, c and d on all tones, the common transmitter is informed about them via reception of CTR. Then the TAS algorithm is run at the C-Tx and the first broadcasting packet informs the receivers about the allocated Tones.

Note that when a, b, c, or d reply to the C-Tx with a CTR message over the control Schannel, all of their neighboring nodes would be able to receive the CTR reply that contains the time reference information of the common transmitter. Therefore, If any of the one-hop neighbors of a, b, c, or d is about to start transmitting data on any of the pre-assigned subchannels, it can adjust its time to the C-Tx to avoid causing multi-transmission loss of orthogonality at a, b, c, or d.

2) Common Receiver (C-Rx):

1) A C-Rx transmits an RTM-R (Request to Multiple Receive) over the control Schannel. The RTM-R contains clock frequency reference, subjected receivers' addresses and time reference and the assigned Schannels.
2) Nodes that successfully receive the RTM-R message transmit a CTS (Clear to Send) message over the allocated Schannel. The CTS message contains pilot data to facilitate fading estimation at the receiver, as well as time reference of the C-Rx node.
3) C-Rx performs channel estimation and runs the tone assignment algorithm and assigns grouping of tones to each neighbor.
4) C-Rx transmits a STS (Start to Send) message on the control Schannel that contains the list of the assigned tones

for each neighbor.

5) Nodes start transmitting data over the assigned tones at the same time according to the clock and time reference of the C-Rx, and C-Rx would be able to separate them based on the assignment.

Figure 3(b) illustrates an example for this scenario when a C-Rx is attempting to receive data from nodes a, b, c and d. In this scenario, to facilitate channel estimation, after reception of RTM-R message, nodes a, b, c and d should send pilots on the assigned Schannels within the CTS message. Meanwhile when one-hop neighbors of a, b, c and d receive the information regarding the time reference of the common receiver, they would be able to make sure that if they are about to be receiving data on any subchannel, the timing of the other transmitter is aligned with the timing used by a, b, c and d. The C-Rx in this case, has to run the TAS algorithm, assign tones to the neighbors, and sends an STS message to inform the neighbors about the assigned tones.

VII. Analysis

A. Analysis of TAS

As indicated, the spectrum in OFDMA is divided into much smaller tones and the channel gain vector $h = [h^1, h^2, ..., h^N]$ includes h^i which represents channel gain for tone i and N represents the number of tones in one OFDMA subchannel. h^i is randomly and independently changing, because the distance between the two adjacent tones is chosen to be smaller than channel coherence bandwidth. These channel gains are highly diverse on different transmission paths, thus intelligent channel assignment could lead to substantial rate improvement [22]. To show how tone selection using TAS can improve data rate, let's assume that $1, 2, ..K$ transmission links are sharing one node as the common transmitter or the common receiver. let $h_k = [h_k^1, h_k^2, ..., h_k^N]$ be the channel gain vector for link k. Note that there are a total of K links that share a common transmitter or receiver (K neighbors for the common node). The channel assignment can be modeled as:

$$H_k = h_k \times \Gamma_k$$

when Γ_k is a $N \times 1$ vector whose elements $\Gamma_k(i) = 1$ if the i'th tone is assigned to link k and zero if not. Then $\Gamma_k \wedge \Gamma_l = 0$, for $k \neq l$, and $\Gamma_1 \vee \Gamma_2 ... \vee \Gamma_K = \mathbf{1}$ when $\vee$ represents logical OR, $\wedge$ represents logical AND. Our goal is to investigate how rate can be improved if $\Gamma_k(i) = 1$ when:

$$|H_k^i|^2 = max(|h_1^i|^2, |h_2^i|^2, ...|h_K^i|^2)$$

Thus, tones in our scheme are assigned in such a way that for each tone (h^i), the link that experiences the highest gain for channel will get to use that tone. We will show how this assignment is carried out fairly, without requiring channel state information at the transmitter side which is a very resource consuming criteria.

Assuming that all tones have the same statistics, the cumulative distribution function of $|H_k^i|^2 = \mathbf{H}_k$ when $|h_k^i|^2 = \mathbf{h}_k$ is:

$$\begin{aligned} F_{\mathbf{H}_k}(x) &= prob(\mathbf{H}_k \le x) \\ &= prob(max(\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_K) \le x) \\ &= prob(\mathbf{h}_1 \; and \; \mathbf{h}_2 \; ...,, \; and \; \mathbf{h}_K \le x) \\ &= \int_{-\infty}^{x} f_{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_K}(y) \; dy \\ &= F_{\mathbf{h}_1}(x) \cdot F_{\mathbf{h}_2}(x) ... \cdot F_{\mathbf{h}_K}(x) = (F_{\mathbf{h}_k}(x))^K \end{aligned}$$

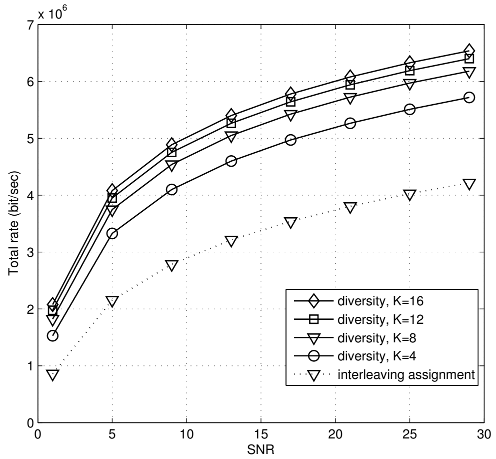


Fig. 4. Total rate versus SNR (P/n) for various number of neighbors (K)

The last equation is derived noting that h_1 to h_K are identically and independently distributed. Therefore, the derivative of the cumulative distribution function (probability distribution function (pdf)) of $\mathbf{H}_k$ is given by

$$f_{\mathbf{H}_k}(x) = K f_{\mathbf{h}_k}(x) \, F_{\mathbf{h}_k}(x)^{K-1}.$$

We assume in this paper that channel is Rayleigh distributed. However, our approach can be extended to any time-varying channel model. The probability distribution function of $\mathbf{h}_k$ is exponential and given by

$$f_{\mathbf{h}_k}(x) = \frac{1}{\sigma} exp^{-x/\sigma}$$

for $x > 0$, when $\sigma = E_{\mathbf{h}_k}(x)$. And the cumulative distribution function is given by

$$F_{\mathbf{h}_k}(x) = 1 - exp^{-x/\sigma}$$

Then the pdf of $\mathbf{H}_k$ can be derived as:

$$f_{\mathbf{H}_k}(x) = \frac{K}{\sigma} \; exp^{-x/\sigma} \; (1 - exp^{-x/\sigma})^{K-1}$$

The total achievable rate on all tones is given by

$$R = \sum_{i=1}^{N} \int_0^{\infty} B/N \; log(1 + x \; P/n) f_{\mathbf{H}_k}(x) \; dx.$$

where B is the bandwidth of a subchannel, P is the transmission power, and n is the noise power. Since we assume equal transmission power on all tones, rate can be rewritten as

$$R = \frac{B \, K}{\sigma} \int_0^{\infty} \; log(1 + x \; P/n) \; exp^{-x/\sigma} (1 - exp^{-x/\sigma})^{K-1} dx.$$

If tones were to be assigned in an interleaving fashion, the distribution function would follow Rayleigh and as a result the rate is calculated by

$$R = \frac{B}{\sigma} \int_0^{\infty} \; log(1 + x \; P/n) \; exp^{-x/\sigma} dx.$$

Figure 4 shows the results of this rate analysis for various K and SNR (P/n) values. It can be observed that more diversity gain can be attained with more communicating links, although the maximum gain is achieved when K increases from 4 neighbors to

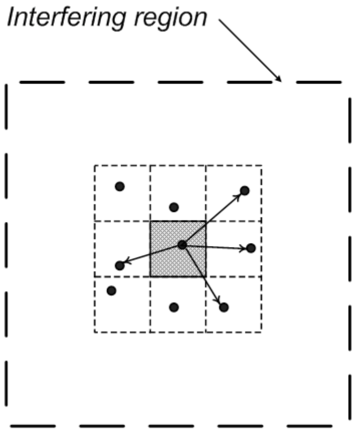


Fig. 5. Model used in the analysis

8 neighbors. It is apparent that diversity gain is more substantial at low SNR, which means that this technique is mostly beneficial in a low SNR environment, longer transmission rates, and weaker receivers.

B. Analysis of CoCo-MAC

To analytically compare the performance of our OFDMA based MAC protocol with traditional multi-channel and single-channel approaches, we assume that there are a total of n nodes randomly distributed on a unit sphere surface that is divided into square cells with area $a(n)$ as described by Gamal, et al. [23]. They prove that, if $a(n) = \Theta(logn/n)$, each cell contains at least one node with very high probability. A node can communicate with any other node in its cell and the 8 neighboring cells and the transmission range is $r(n) = \sqrt{8a(n)}$ to ensure connectivity. Based on the relaxed protocol model [24], a successful transmission of data from node i to node j on a specific Schannel is possible if for any other node k transmitting data on the same Schannel:

$$d_{kj} = (1 + \Delta)r(n)$$

where d_{kj} represents the distance from node k to node j. The *guard interval* Δ has a direct relationship with the minimum signal-to-interference ratio (SIR) necessary for the physical layer to successfully receive data bits. Therefore, two transmitters sending data on the same Schannel should be at least $(2+\Delta)r(n)$ away from each other. As a result, for a node located in a cell, any other node located in a square with side of $(4.35 + 2\Delta)r(n)$ surrounding this node could be a potential interferer. We focus on the performance of the nodes located inside one interfering region because the performance of MAC primarily is bounded by the behavior of the protocol in the interfering neighborhood. Fig 5 shows a cell, its eight neighboring cells and the interfering region. The number of cells in an interfering region is found to be:

$$N_{ce} = \frac{[(4.35 + 2\Delta)r(n)]^2}{a(n)} = 8(4.35 + 2\Delta)^2$$

$N = N_{ce}na(n)$ is the number of nodes in this region. Nodes try to attempt the channel with rate $p = K \times \lambda$ to communicate with K of their one-hop neighbors when λ is the packet arrival rate for one neighbor. We also assume that the time is slotted with perfect synchronization at slot boundaries.

1) CoCo-MAC: The behavior of the MAC protocol is modeled using a Markov chain when the packet length is assumed to be geometrically distributed with parameter q and the average packet length is $L = 1/(1-q)$. At any given point in time, the state i is represented by (k_i, l_i) where k_i is the number of nodes receiving data and l_i is the number of nodes transmitting data in the region. we can numerically calculate the transition probabilities $P_{(ij)}$ from state i to state j when states are numbered from 1 to M, and π_i represents the steady state probability of state i. To find π_i for all i, we need to solve the global balance equations as follows:

$$\sum_{i=1}^{M} \pi_i = 1 \ \ , \ \pi_j = \sum_{r=1}^{M} \pi_r P_{(rj)} \ \ for\ j = 1\ to\ M$$

Throughput or the average number of packets received per time slot is basically the average number of communication links. In CoCo-MAC, C-Tx and C-Rx establish communication via the transmission of RTM-S or RTM-R. Both forms of negotiations result in the same probability of setting up new communication links. Also, terminating of a link only depends on the average packet length. Therefore, we calculate throughput considering one form of negotiation only, which is C-Tx. The Throughput then can be calculated as:

$$S = \sum_{i=1}^{M} k_i \pi_i$$

where k_i is the number of receivers or communication links in state i. The length of each time slot is assumed to be equal

$$\max\left[(T_{(RTM-S+Pilot)} + 2\sigma), (K \times T_{(CTR+\sigma)})\right]$$

where σ is the propagation time for a packet and T is the transmission time for a packet. The maximum number of achievable communication links is bounded by the number of nodes in the region and the number of available Schannels, N^S, $l_{max} = Min(N^S, \lceil Nce/9 \rceil \times K)$.

A transition from one state to another takes place if a new RTM-S is successfully received by one or multiple idle nodes, or an existing active link between a transmitter and a receiver is terminated. The state transition probability p_{kl}^{mn} from state (k, l) to state (m, n) can be expressed as follows:

$$p_{kl}^{mn} = \sum_{(\alpha,\beta,\mu,\eta)\in S_t} \left\{R_k(\alpha)\ \cap T_{k,l}(\mu|\alpha)\ \cap RT_{k,l}(\eta, \beta|N^a)\right\} \quad (2)$$

where $R_k(\alpha)$ is the probability that α links are terminated when k receivers are actively receiving data in the region:

$$R_k(\alpha) = \binom{k}{\alpha}(1-q)^{\alpha} q^{k-\alpha} \quad (3)$$

and $T_{k,l}(\mu|\alpha)$ is the probability that μ transmitters stop transmitting to all of their receivers given that α links have been terminated. Based on our analysis and its comparison with the simulation's results, the probability distribution function of μ is expressed as:

$$T_{k,l}(\mu|\alpha) = \mu^{(\kappa-1)} \frac{e^{\mu/\theta}}{\theta^{\kappa}\,\Gamma(\kappa)}$$

Assuming μ is gamma-distributed, $\mu \sim \Gamma(\kappa, \theta)$, with shape parameter $\kappa \propto \alpha$, and scale parameter $\theta \propto l$.

$RT_{k,l}(\eta, \beta|N^a)$ is the joint probability that η new transmitters start to transmit data and β new receivers start to receive data given that $N^{'}$ nodes are available in the new state. Only one successful RTM-S can be transmitted in the interfering region. Therefore, only one new transmitter can be added moving to a new state. There are a total of $l_{av} = N^S - (k - \alpha)$ links left to be used in the region. To add β new receivers, this request must be aimed for β idle neighboring nodes and there needs to be enough Schannels assigned to the transmitter. Accordingly, β cannot be more than l_{av}. Also β cannot be larger than the average number of nodes' one-hop neighbors, N_n, and also K when $K \leq N_n$. N^a is the total number of available nodes and is equal to $N - ((k + l) - (\alpha + \mu))$.

$$RT_{k,l}(\eta, \beta | N^a) =$$

$$\begin{cases} \delta(\eta - 1)\, T_e\, (\frac{N_n^a}{N_n})^{\beta} + \\ \delta(\eta)\, ((1 - T_e) + & if\ \beta \leq l_{av},\ \beta \leq N_n; \\ T_e\, (1 - (\frac{N_n^a}{N_n})^{min(m,N_n)}) \\ \\ 0 & if\ \beta > l_{av},\ \beta > N_n; \end{cases}$$

T_e is the probability that one successful RTM-S has been transmitted in the interfering region:

$$T_e = \binom{N^a}{1} p(1 - p)^{N^a - 1} \tag{4}$$

Note that N_n^a is the average number of idle nodes among the nodes' one-hop neighbors: $N^a \cdot N_n / N$.

S_t represents all possible values for the set $(\alpha, \beta, \mu, \eta)$ when $l = 0\ to\ (l_{max}+1)/2$, $k = l\ to\ (l_{max}+l-l)$, and $n = 0\ to\ (l_{max}+1)/2$, $m = n\ to\ (l_{max} + 1 - n)$.

$$S_t = \begin{cases} (\alpha,\ (m - k) + \alpha,\ (l - n) + \eta,\ \eta) : & if\ n{-}l{\leq}1, \\ \forall\ \alpha,\eta{\in}N,(1{\leq}\alpha{\leq}k,\ \eta{=}1) & m{-}k{\geq}1; \\ \\ ((k - m) + \beta,\ \beta,\ (l - n) + \eta,\ \eta) : & if\ n{-}l{\leq}1, \\ \forall\ \eta,\beta{\in}N,(\eta{=}0,\ \beta{=}0) or (\eta{=}1,\ 1{\leq}\beta{\leq}m) & m{-}k{<}1; \\ \\ 0 & if\ n{-}l{>}1; \end{cases}$$

After finding S_t depending on the value of (k, l) and (m, n), we can numerically calculate the transition probabilities. States are numbered from 1 to $M = \frac{l_{max}(l_{max}+1)}{2} + 1$. Then after solving the global balance equations, throughput as the average number of packets received per time slot can be calculated.

2) FDMA Channel Switching MAC: We compare the performance of CoCo-MAC with a generalized MAC protocol that uses a dedicated control channel.

In a MAC based on a dedicated control channel, a node transmits an RTS on the common control channel. The receiver of the RTS replies with a CTS that includes the agreed channel number and both the transmitter and the receiver switch to the channel to carry on the communication. An example of such protocol is MMAC [2] which is a multi-channel MAC based on exchanges of RTS-CTS over a control channel. Given that each node can communicate with only one of the nodes in a neighboring cell, the maximum number of achievable simultaneous links is bounded by:

$$l_{max} = Min(c, \lceil Nce/9 \rceil . \frac{N_n + 1}{2})$$

where c is the number of available orthogonal data channels. At any given point in time, the state of the system is represented by the number of pairs of nodes engaged in data transmission, denoted by k. The length of a time slot equals the length of an RTS plus CTS plus four propagation delays. To find the transition probability, $R_k(\alpha)$ is the probability that α links are terminated, as shown in Eq. 3. The probability of adding η links in the new state, $T_k(\eta | N^a)$, is found to be:

$$T_k(\eta | N^a) =$$

$$\begin{cases} \delta(\eta - 1)\, T_e\, (\frac{N_n^a}{N_n}) + & if c - (k - \alpha) \geq 1; \\ \delta(\eta) \Big((1 - T_e) + T_e\, (1 - (\frac{N_n^a}{N_n}))\Big) \\ 0 & if\ c - (k - \alpha) < 1; \end{cases}$$

where $N^a = N - (2k)$ represents the total number of available nodes. T_e is calculated as Eq. 4 shows. The transition probability from state k to state m is:

$$P_k^m = \sum_{(\alpha,\eta) \in S_t} R_k(\alpha).T_k(\eta | N^a)$$

The space S_t includes all the possible values for the set (α, η) when $\{k = 0\ to\ l_{max}\}$ and $\{m = 0\ to\ l_{max}\}$:

$$S_t = \begin{cases} ((k - m) + \eta,\ \eta) : \forall\, \eta{\in}N,(0{\leq}\eta{\leq}1) & if\ m - k \leq 1; \\ 0 & if\ m - k > 1; \end{cases}$$

3) 802.11 collision avoidance: Because a single channel is available, the number of active links in the interfering region can be either 0 or 1. Therefore, the Markov chain can have two possible states. Given that each node can use the entire bandwidth compared to CoCo-MAC, the average packet length is equal to $L = \frac{1}{2(N^S)(1-q)}$ and for the same L, q is increased to $1 - \frac{1}{LN^S}$.

VIII. Analysis and Simulation Results

Qualnet [25] simulations were carried out to verify SSA and CoCo-MAC. In these simulations, 50 nodes were distributed uniformly in an area that changed depending on network degree (Ne). Since Qualnet does not offer bit-level simulations, we used MATLAB to evaluate TAS and find the average achievable bit rate per transmission link, and used that information to obtain the average throughput.

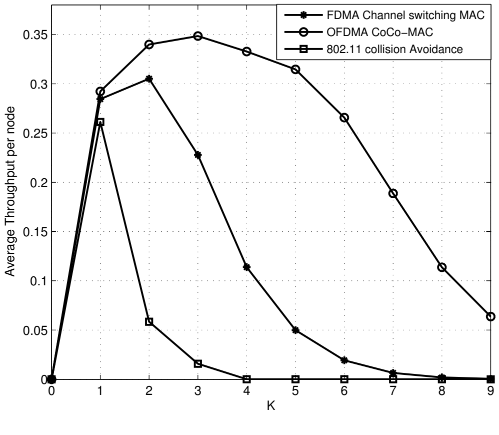
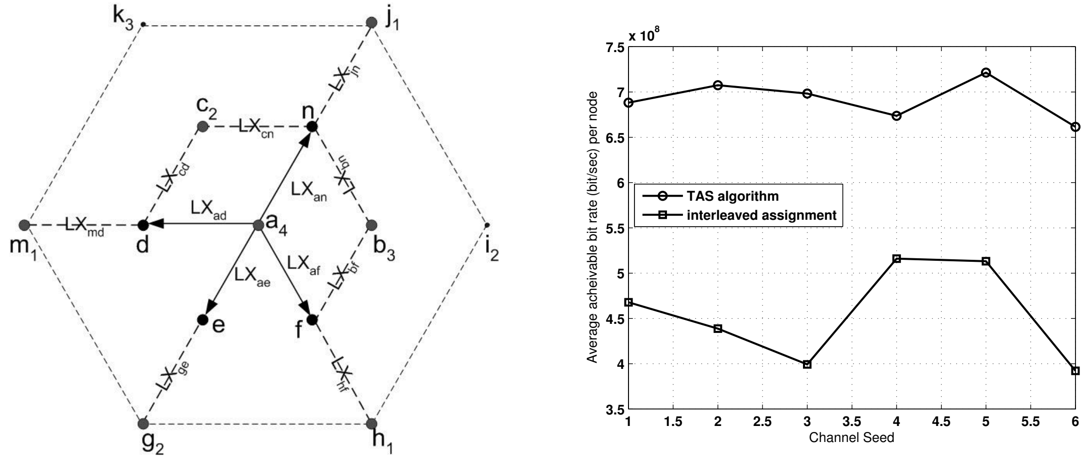


Fig. 7. Analysis results for average throughput per node



(a) Network topology

(b) Average achievable bit rate per node sent from node a for various channel seeds

Fig. 6. TAS MATLAB simulation

A sample topology in MATLAB is shown in Figure 6(a). In this case, node a is a common transmitter and is attempting to send data to nodes n, f, d and e. SSA had assigned groups of Schannels to each node. We give a channel number to each group, and channel 4 is the channel for node a. Each link is modeled to be facing Rayleigh distributed channel. Random seeds were selected to simulate the rate under different fading channel conditions.

Figure 6 presents the average achievable bit rate per node a's receivers for various channel seeds when SNR is equal to 10 and 2k tones are available. To observe the improvement; the same bit rate is plotted when tones are assigned in an interleaved fashion to the receivers without considering fading. The average ratio of improvement for 50 seeds is 1.42. The analysis we did in Section VII-A showed that in average when number of receivers is equal to 4, and $SNR = 10$, the improvement ratio should be about 1.45. The difference here clearly is due to the randomness of the simulations in spite of using many random seeds.

In Qualnet simulations, packets have MTU of 512B long, and we averaged our experimental values over 10 different random stationary topology. The signal attenuation is assumed to be based on the channel model according to an indoor environment by delay spread of 200ns.

Figure 8 illustrates the percentage of nodes with a confirmed Schannels resulting from the hashing stage with $K = 1$ versus average network degree when various number of Schannels, N^S, are available. As it can be observed, as the network degree increases (dense networks), the hashing stage can result in high percentage of confirmed Schannels, especially when many Schannels are available (which is the case in OFDMA networks). With high percentage of confirmed Schannels, SEQ can distribute the rest of the Schannels with minimum overhead.

Figure 11 shows the throughput per node in bit/sec versus K for CoCo-MAC when TAS algorithm is utilized and a version of CoCo-MAC that only uses concurrency and adaptive subchannel allocation (no diversity). Also, the results are compared with MMAC [2] which is also a multi-channel MAC based on exchanges of RTS-CTS over a control channel but only utilizes a single orthogonal channel per transmission (same overall BW as CoCo-MAC for the sake of comparison). Figure 7 illustrates the results of our analysis via MATLAB. In this example, $\delta = 0.4$, and $L = 3$ when number of nodes is 50 and $N_e = 10$. As we discussed in Section IV, our subchannel assignment algorithm, SSA, adaptively distributes Schannels within each two-hop neighborhood based on the traffic requirements. Hence, as the number of communicating neighbors increases, the algorithm assigns more Schannels to the node. However, with a channel switching network, only a fixed number of channels can be utilized. This

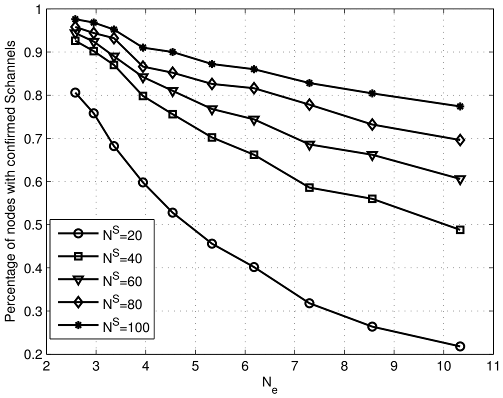


Fig. 8. Total number of confirmed Schannels after the hashing stage

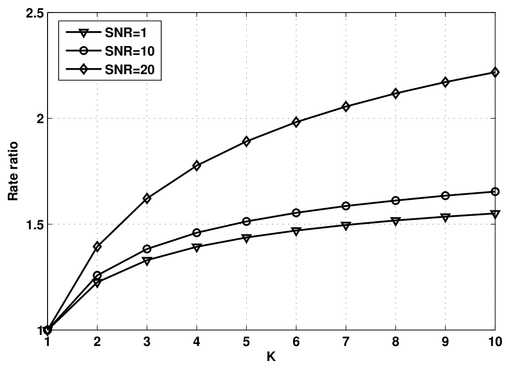


Fig. 9. Rate ratio of TAS to random selection technique for various values of K and SNR

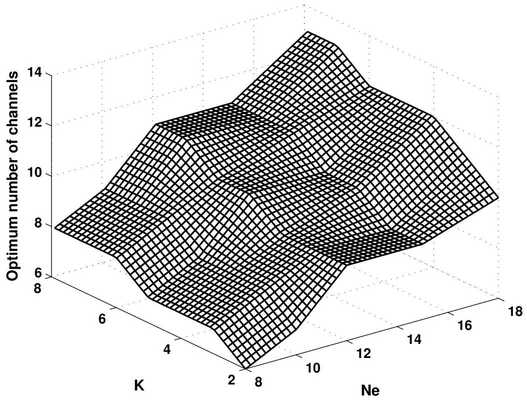


Fig. 10. Optimum number of channels versus Network degree (Ne) and K

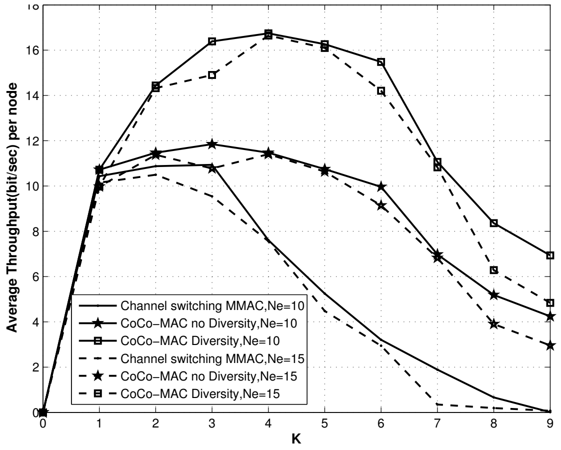


Fig. 11. Throughput results for various K and network degree

affects the throughput as K is increasing. As shown in Figure 11, as K goes up from 0 to 3, the throughput increases, because more communication is taking place in the network. However, as K increases, with MMAC, the number of channels is not enough to support the traffic, and due to contention, throughput declines. In this simulation, the number of channels is equal to 5. The fewer the channels, the higher the throughput at lower values of K but declines faster. In CoCo-MAC, the number of channels is considered to be $N_C = N^S/N_C^S$, where N_C^S is the average number of Schannels assigned to each node. We discussed how N_C is adaptive and although contention is present, the reduction of throughput is much slower than with a channel switching MAC protocol. Figure 10 shows the number of channels that result in best performance for various number of K and Ne. The results from our simulations are very similar to what we observed from our analysis, as seen in Figure 7.

When TAS is utilized as described in the previous section, diversity results in higher throughput. As shown in Figure 9, the ratio of achievable rate using TAS to an interleaved technique has a sharp increase and slows down after $K = 6$ when $SNR = 10$. This is because diversity takes advantage of the difference between multiple channels and this difference will plateau eventually. As shown in Figure 11, CoCo-MAC with diversity gives a great performance as traffic demands increases and has higher bit rate at all traffic needs due to diversity.

IX. Conclusion

We presented a new cross-layer channel allocation technique for OFDMA ad hoc networks. Previous cross-layer MAC protocols for ad hoc networks fail to adapt OFDMA at the physical layer and exploit its advantages in terms of multiuser diversity. The novelty of this work is a tone-assignment algorithm that takes advantage of channel fading to improve data rate through multiuser diversity and a MAC protocol that addresses the synchronization requirements of OFDMA and the tone assignment algorithm's necessities while enabling concurrent initiation of data transmissions. The results from our simulations show that the achieved improvement is due to concurrency, the added diversity rate, and the adaptive allocation of BW to nodes.

References

[1] J. Mo, H. W. So, and J. Walrand, "Comparison of multichannel mac protocols," *Proc. ACM MSWiM05*, 2005.

[2] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *MobiHoc '04*, 2004, pp. 222–233.

[3] J. S. E.S. Sousa, "Spreading code protocols for distributed spread-spectrum packet radio networks," *IEEE Transactions on Communications*, vol. 36, pp. 272–281, Mar. 1988.

[4] K. Xing, X. Cheng, L. Ma, and Q. Liang, "Superimposed code based channel assignment in multi-radio multi-channel wireless mesh networks," in *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, New York, NY, USA, 2007, pp. 15–26.

[5] P. Kyasanur and N. Vaidya, "Capacity of multi-channel wireless networks: impact of number of channels and interfaces," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking.* ACM, 2005, pp. 43–57.

[6] N. Jain, S. Das, and A. Nasipuri, "A multichannel csma mac protocol with receiver-based channel selection for multihop wireless networks," *Proc. IEEE IC3N*, 2001.

[7] *IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE Std., 2002.

[8] *Digital Video Broadcasting (DVB); Interaction channel for Digital Terrestrial Television (RCT) incorporating Multiple Access OFDM (MAC) and Physical Layer (PHY) specifications*, ETSI Std., 2002.

[9] D. Niyato and E. Hossain, "Adaptive fair subcarrier/rate allocation in multirate ofdma networks: Radio link level queuing performance analysis," *Vehicular Technology, IEEE Transactions on*, vol. 55, no. 6, pp. 1897–1907, Nov. 2006.

[10] I. Stiakogiannakis, D. Zarbouti, G. Tsoulos, and D. Kaklamani, "Subcarrier allocation algorithms for multicellular ofdma networks without channel state information," in *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, May 2008, pp. 73–77.

[11] Y. Peng, S. Armour, and J. McGeehan, "An investigation of dynamic subcarrier allocation in mimoofdma systems," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 5, pp. 2990–3005, Sept. 2007.

[12] B. Bai, W. Chen, Z. Cao, and K. Letaief, "Achieving high frequency diversity with subcarrier allocation in ofdma systems," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, 30 2008-Dec. 4 2008, pp. 1–5.

[13] G. Kulkarni and M. Srivastava, "Subcarrier and bit allocation strategies for ofdma based wireless ad hoc networks," *GLOBECOM '02. IEEE*, vol. 1, pp. 92–96 vol.1, 2002.

[14] H. Li, "Ofdma resource allocation in hybrid wireless network," in *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, sept. 2009, pp. 1 –5.

[15] V. Venkataraman and J. J. Shynk, "Adaptive algorithms for ofdma wireless ad hoc networks with multiple antennas," vol. 1, 2004, pp. 110–114 Vol.1.

[16] H. Shetiya and V. harma, "Algorithms for routing and centralized scheduling in ieee 802.16 mesh networks," *WCNC 2006. IEEE*, pp. 147–152, 2006.

[17] S. Valentin, T. Freitag, and H. Karl, "Integrating multiuser dynamic ofdma into ieee 802.11 wlans - llc/mac extensions and system performance," *Communications, 2008. ICC '08. IEEE International Conference on*, pp. 3328–3334, May 2008.

[18] S. Barbarossa, M. Pompili, and G. Giannakis, "Channel-independent synchronization of orthogonal frequency division multiple access systems," *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 2, pp. 474–486, Feb 2002.

[19] M.-O. Pun, M. Morelli, and C.-C. Kuo, "Maximum-likelihood synchronization and channel estimation for ofdma uplink transmissions," *Communications, IEEE Transactions on*, vol. 54, no. 4, pp. 726–736, April 2006.

[20] H. Yaghoobi, "Scalable ofdma physical layer in ieee 802.16 wirelessman."

[21] U. H. L. Kivanc, D.; Tureli, "Fair resource allocation in an uplink ofdma system," *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pp. 1695–1699, 2007.

[22] T. Rappaport, *Wireless Communications - Principles and Practice.* Prentice Hall, 2001.

[23] A. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," *In Proc. 2004 INFOCOM*, 2004.

[24] P. Gupta and P. Kumar, "Capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 2, pp. 388–404, March 2000.

[25] S. N. Technologies, "Introduction to qualnet, qualnet user's manual," 2004.