This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Distributed coordination with deaf neighbors: efficient medium access for 60 GHz mesh networks

Sumit Singh*, Raghuraman Mudumbai† and Upamanyu Madhow*
*Dept. of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106, USA
†Dept. of Electrical and Computer Engineering, The University of Iowa, Iowa City, IA 52242, USA

***Abstract*—Multi-gigabit outdoor mesh networks operating in the unlicensed 60 GHz "millimeter (mm) wave" band, offer the possibility of a quickly deployable broadband extension of the Internet. We consider mesh nodes with electronically steerable antenna arrays, with both the transmitter and receiver synthesizing narrow beams that compensate for the higher path loss at mm-wave frequencies, achieving ranges on the order of 100 meters using the relatively low transmit powers attainable with low-cost silicon implementations. Such highly directional networking differs from WiFi networks at lower carrier frequencies in two ways that have a crucial impact on protocol design: (1) directionality drastically reduces spatial interference, so that pseudowired link abstractions form an excellent basis for protocol design; (2) directionality induces deafness, which makes medium access control (MAC) based on carrier sensing infeasible. Interference analysis in our prior work shows that, in such a setting, coordination between transmitters and receivers, rather than interference management, becomes the key MAC performance bottleneck. However, the question of whether such coordination can be achieved in a *distributed* fashion while achieving high medium utilization, was left open. In this paper, we answer this question in the affirmative, presenting a distributed MAC protocol that employs memory to achieve approximate time division multiplexed (TDM) schedules without explicit coordination or resource allocation. The efficacy of the protocol is demonstrated via packet level simulations, while a Markov chain fixed-point analysis provides insight into the effect of parameter choices.**

I. Introduction

The US Federal Communications Commission (FCC) has set aside 7 GHz of unlicensed spectrum in the 60 GHz band, which is a part of the broader "millimeter (mm) wave" band. Mm wave spectrum has been used in military communications and radar systems for decades (with RFICs based on expensive packaging techniques and compound semiconductor processes). However, it is now becoming commercially attractive given the recent successes in building mm-wave transceivers in inexpensive silicon processes, using low-cost packaging techniques [1], [2]. While there is significant industry interest in indoor wireless personal and local area network applications of 60 GHz networking, our focus here is on multiGigabit *outdoor* 60 GHz mesh networks with link ranges on the order of 100 meters. Such networks, possibly in conjunction with longer range wireless or fiber links, can provide a quickly deployable broadband infrastructure at locations where it is inadequate or absent, as well as serve as a means of stimulating competition by enabling smaller broadband service providers to bypass the existing infrastructure provided by entrenched carriers. Oxygen absorption in the 60 GHz band, which makes long-range (kilometers) links infeasible, plays a relatively insignificant role in our link budget: at a range of 100-200 meters, Oxygen absorption loss at 15 dB/km [3] contributes only 1.5-3 dB. However, Oxygen absorption does help in spatial reuse by significantly attenuating interference from far-away nodes.

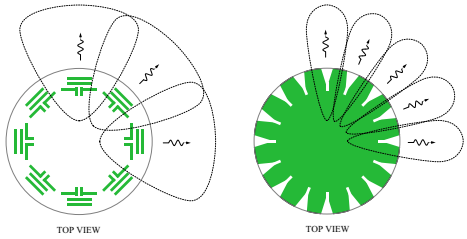
In this paper, we address what we believe is a critical bottleneck in the design of 60 GHz mesh networks as described above: medium access control (MAC) for highly directional links. We constrain the links to be highly directional because it is both *essential* and *feasible.* Directionality is *essential* because, for omnidirectional transmission, the path loss scales as λ^2, where λ is the carrier wavelength, so that 60 GHz is 22 dB worse than 5 GHz. Given the difficulty of producing RF power at mm-wave frequencies, such a loss in the link budget is unacceptable. By using narrow beams at both the transmitter and receiver, however, we can more than compensate for the loss: for the same antenna size, the directivity scales as λ^{-2}, so that the overall path loss with directive antennas at both ends scales as λ^{-2}, leading to a net *gain* of 22 dB over 5 GHz. Directionality is *feasible* because, at these small wavelengths, large-scale antenna arrays can be realized as patterns of metal on circuit board, and can be employed to synthesize electronically steerable (e.g., see [4], [5]), highly directional beams for nodes with form factors comparable to that of a WiFi access point.

The use of highly directional links requires a complete rethinking of MAC protocol design relative to CSMA-based WiFi networks for two reasons. Since spatial interference is significantly reduced, interference management or avoidance becomes a secondary consideration. On the other hand, the deafness resulting from directionality means that carrier sensing or promiscuous listening in order to monitor the state of one's neighbors is no longer feasible. Indeed, in our prior work [6], we showed that, for the kinds of link directivities we consider, collision probabilities are small even with transmitters that are completely oblivious of each other, and that, for a naïve slotted Aloha protocol, packet losses due to failure of transmitter-receiver coordination are an order of magnitude larger than those from interference. The challenge that we take up in this paper is whether we can design a *distributed* MAC protocol that is scalable and robust and that achieves coordination in the face of deafness, for efficient medium utilization.

Before discussing network design, we note that electronically steerable links can be realized for easy-to-deploy mesh nodes that can connect with a neighbor node regardless of the direction of the neighbor's location. Several concept

978-1-4244-5837-0/10/$26.00 ©2010 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.



(a) Yagi-Uda antenna array (b) Open-slot antenna array

Fig. 1. Steerable arrays with high directivity and compact form factors.

designs for steerable arrays with a 360° field of view for such *omni-coverage yet highly directional* nodes are currently being explored: Figs. 1(a) and 1(b) depict two representative array implementations with Yagi-Uda and open-slot antennas, respectively. Each element in the array can have significant directivity (e.g., up to 20 dBi), so that such arrays can be operated in beamswitched mode, but even higher directivity (e.g., up to 26 dBi) can be obtained by combining signals from a small number of elements with overlapping fields of view. While hardware optimizations for efficient realization of baseband and RF processing at such nodes are topics of ongoing research, the hardware developed for emerging indoor 60 GHz products [1], [2] implies that there are no fundamental roadblocks in realizing such nodes.

Contributions: We present a *distributed* MAC protocol that attains high medium utilization, comparable to efficient time division multiplexing (TDM), while avoiding the network-wide explicit coordination required to implement global TDM schedules. The key idea is to use memory to achieve implicit coordination among the mesh nodes despite deafness. Each node persists in using the transmit or receive slots that have been successfully used for a given neighbor, and puts slots over which it has been unsuccessful on a per-neighbor blacklist. There is no attempt at proactive interference avoidance and the blacklisting is oblivious to whether the failure was due to deafness (more likely) or interference (less likely). Thus, a node's transmit and receive history with each of its active neighbors provides feedback that is used for implicit coordination, and persistent use of a given slot for transmitting to a given neighbor leads to an approximate TDM schedule. The system is prevented from locking into undesirable schedules by each node leaving "enough room" in its schedule, along with suitable randomization of persistence and blacklist lifetimes, for adaptively accommodating changes in demand.

The novelty of our approach lies in the use of simple learning rules employed by each mesh node, based *only* on the memory of its own transmit/receive outcomes, to converge to TDM-like schedules in a distributed fashion with minimal control overhead. In addition to satisfying the design constraints unique to 60 GHz mesh networks, this approach obviates the need for frequent control message exchanges for coordination or local neighborhood information gathering/dissemination typical of the prior works on distributed MAC scheduling. The advantages over a centralized approach are also worth stating: lower overhead, and better scalability and fault tolerance.

We offer approximate analytical insights into the working of our protocol via a Markov chain fixed-point analysis, and study the properties of the approximate TDM schedules that arise from our protocol. We present an extensive evaluation of our protocol using the QualNet network simulator [7], modified to model 60 GHz communication.

Related Work: To the best of our knowledge, this is the first work on distributed MAC protocol design for mm-wave outdoor mesh networks, following up on the design considerations brought out by our prior work [6] on interference analysis for such networks. We note that past research on MAC design for 60 GHz networks has focused exclusively on indoor personal and local area networks. This includes our own work on indoor mm-wave networks controlled by an access point [8], where the goal was to employ multihop communication to deal with link blockage.

Directional networking for multihop wireless networks in the WiFi band has been extensively studied in the past [9]–[15]. Many of these proposals employ directional networking as an option to enhance the network performance (via high spatial reuse or range extension) while still exploiting the broadcast nature of omnidirectional communication for critical control message propagation. Thus, such protocols do not apply to 60 GHz mesh networks where there is no omnidirectional mode. Further, most directional protocols designed for lower frequency bands consider interference management as a prominent MAC design goal and use explicit signaling for proactive interference avoidance, since the achievable antenna directivities are much smaller than those possible at 60 GHz.

A few protocols that employ an out-of-band tone to compensate for the loss of coordination [12]–[14] require additional control and hardware complexity: [13], [14] need multiple simultaneously active transceivers to concurrently transmit a directional out-of-band busy-tone, whereas [12] requires delayed transmission of an omnidirectional out-of-band tone. A variation of slotted Aloha is proposed in [15], where each idle node equipped with an adaptive antenna array beamforms towards the strongest tone (tones precede every packet transmission), and forms nulls towards potential interferers. Another approach is to use directional flooding of control messages on all sectors [11]; this would have prohibitive overhead for 60 GHz nodes employing a large number of narrow sectors.

While our goal here is to design a MAC protocol providing high network utilization and a rough measure of fairness, the pseudowired abstraction presented here can also form the basis for scheduling algorithms (perhaps layered as a small perturbation on top of a MAC such as ours) that explicitly account for traffic patterns. This is an important topic for future work, especially for providing the QoS that may be required for a wireless backhaul network. We anticipate that the large body of work on distributed link scheduling can be leveraged for this purpose [16]–[18].

Note that we have previously employed the idea of "sticking" to TDM-like schedules in a completely different context: efficient support of periodic traffic in omnidirectional CSMA networks [19]. In [19], transmissions stick to periodic

schedules after a successful initial handshake, and use the memory of carrier-sensed activity to avoid contention. Related work includes [20], which uses two priority level broadcast control messaging over each slot, guided by the memory of the past transmission outcomes. While the application of memory in this paper is quite different (enabling coordination in the face of deafness), all these works indicate that memory and learning could potentially be powerful (and presently underused) tools in network protocol design.

II. 60 GHz Outdoor Mesh Architecture

We envision a rooftop or lamp-post based network with line of sight (LoS) links. Our baseline scenario, used throughout this paper, is a 2 Gbps link at a range of 100 meters. Assuming Quadrature Phase Shift Keying (QPSK) signaling (2 bits/symbol) at 1 Gsymbol/sec over a bandwidth of 1.5 GHz, for a desired signal to noise ratio (SNR) of 15 dB (which allows for uncoded error probability of 10^{-9} or better), Oxygen absorption loss at 15 dB/km, and 10 mW transmit power, we need antenna gains of about 24 dBi at both the transmitter and the receiver in order to provide a 10 dB link margin.

In addition to the half duplex constraint, we assume that each node can only communicate with a single neighbor during a time-slot: all the other signals are treated as interference. Techniques such as multiuser detection and interference nulling could potentially enhance network performance, but our objective here is to explore the performance with a simple physical layer implementable in the short term.

While we focus on MAC design here because it presents the greatest challenge, requiring adaptation at the packet time scale, we comment briefly on possible approaches for slower time scale mechanisms such as network discovery and slot-level synchronization, which the MAC protocol builds on. We assume that nodes obtain information about their neighbors as well as the antenna array weights to communicate with them via a neighbor discovery procedure executed during network initialization. For example, a distributed variant of the algorithm proposed in [8] can be employed: a discovering node sequentially scans each sector (angular region determined by the antenna beamwidths) for the presence of neighbors by sending a hello message and waits for the response(s). The hello message payload is preceded by a preamble that acts as a training sequence for the receiving nodes to beamform in the direction of the transmit node. The beamformed antenna array weights obtained by the receiving nodes are recorded and used while responding back, using reciprocity. Unregistered neighbors that successfully receive a hello message respond within a time window, with a random transmit start time delay to avoid a collision at the discovering node. This procedure is repeated multiple times such that all neighbors are registered, where the hello message sent over the last round requires any unregistered node to respond immediately to verify that no neighbor is left unregistered. The order in which nodes perform network discovery can either be coordinated by the gateway nodes, or it can be configured by the network administrator. Note that the execution of the discovery procedure for the complete network does not take more than a few seconds, which is acceptable for bootstrapping the network.

We assume a time-slotted system where each node is coarsely synchronized with its immediate neighbors, i.e., the slot boundaries are assumed to be aligned to within a guard time. We define a frame to be a sequence of k contiguous time-slots. While the frame duration is assumed to be common, the frame start times or the time-slot indexing may be different for different nodes. In principle, a coarse time synchronization can initially be achieved during network discovery and re-calibrated thereafter, via the gateway nodes. Note that for outdoor mesh nodes, GPS-based synchronization is also a viable option. While a detailed investigation of distributed or gateway-led synchronization mechanisms is beyond the scope of this paper, we note that effective local synchronization mechanisms that account for the high link directionality can be devised, for example, using distributed approaches inspired from [21], [22].

We assume that after network initialization, a receiving node has the ability to "tune into" a transmission from any of its neighbors. Such capability, which greatly simplifies MAC design, does not necessarily require advanced signal processing. For example, a receiver can rapidly scan its antenna beam in the (known) directions of all of its neighbors at the beginning of a time-slot to look for possible incoming transmissions. In this manner, the signal acquisition and beamforming overhead can be kept reasonably low.

A complete network architecture also requires design and optimization of the higher layers, including possibly admission control, routing, and traffic-sensitive scheduling, in addition to MAC. A detailed investigation of these issues is beyond the scope of this paper. In our MAC performance evaluations, we restrict attention to two scenarios: saturated traffic (to evaluate medium utilization efficiency) and constant bit rate traffic with static routing (to evaluate delay and delay jitter performance).

III. Memory-guided Directional MAC (MDMAC)

By virtue of deafness, the only feedback immediately available to a node is regarding its own transmitted and received packets. The novelty of our protocol lies in using memory and learning to arrive at approximate TDM schedules, based on this minimal information. Given the reduced spatial interference, nodes do not attempt to account for the effect of their own transmissions on their neighbors. However, mechanisms for adapting the TDM schedules are built in; this allows reaction to interference (in the few cases where it does occur) and changes in the traffic pattern, and avoids locking into grossly unfair schedules. We begin our exposition with a naïve approach to using memory, and successively introduce the refinements that lead to our final design, in order to clarify the reasoning behind our design choices.

A. Use of Memory for Implicit Coordination

Suppose that node A wishes to transmit to a neighbor node B, and randomly picks one of the free slots in a frame to do so. If the transmission is successful, node B responds with an

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

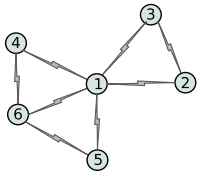


Fig. 2. A toy example.

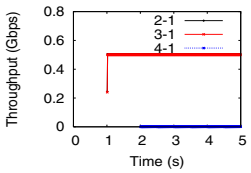


Fig. 3. Node 4 suffers outage.

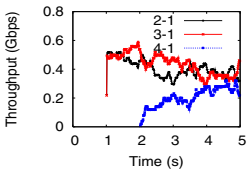


Fig. 4. Prob. state reset.

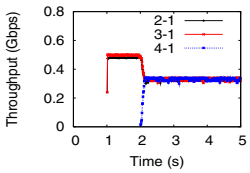


Fig. 5. Explicit state reset.

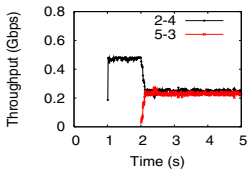


Fig. 6. Two hop flows.

ACK, and both nodes mark the slot as designated for communication from A to B in future frames, thus creating an *implicit reservation.* Node B uses the receive beamforming weights learnt during the first transmission to beamform towards A at the beginning of future slots corresponding to this reservation, and uses reciprocity to transmit beamform its ACKs to node A. The A→B reservation persists over multiple frames until (1) A has no more packets to send to B, or (2) A's transmission fails repeatedly over multiple frames, or (3) A or B explicitly terminates the reservation. If A's initial transmission to B fails (i.e., A does not receive an ACK from B), it flags the slot as "blocked" for future transmission attempts to B.

Problems with the naïve approach: Clearly, if all nodes follow the preceding procedure, the network settles into a TDM schedule without explicit coordination. However, this naïve approach can have a number of problems. Nodes which start late can get locked out in a saturated network, and deadlock conditions can arise due to, for example, nodes A and B trying to send to each other in the same slot. As a toy example, consider the simple six-node network in Fig. 2, where nodes 2 and 3 each have a high data rate flow to node 1 starting at 1s. Node 4 attempts to obtain bandwidth starting at 2s. Assume that all the flows have a base rate enough to saturate the 1 Gbps link. We simulate this scenario over QualNet, using simulation parameters described in detail later, in Section V. Fig. 3 shows the instantaneous flow throughputs. While the throughputs for the flows 2→1 and 3→1 settle to half the link capacity, flow 4→1 gets completely locked out because we have not yet put in mechanisms for perturbing the TDM schedules to accommodate new flows.

Probabilistic State Reset: We now introduce a decentralized, probabilistic mechanism that produces enough churn to allow rearrangements in the TDM schedules. We limit the slot state lifetimes in the following manner: over each frame, there is a nonzero probability that a slot state (e.g., transmit to, receive from, or blocked for transmit attempts to, a given neighbor) is reset, making the slot available for future frames. Thus, each node gives up some of its committed transmit or receive slots and also forgets the blocked slot information after a random number of slots. Due to the randomization in this mechanism, different slots become available at different times, thereby offering all the neighbor nodes a chance to grab these slots. The reset probabilities can be state-dependent, and can be tuned to respond to the expected traffic dynamics.

For our simple example of Fig. 2, we find that assigning random lifetimes to slot states (e.g., with average lifetime of 100 frames) does prevent node 4 from being completely locked out. However, as shown in Fig. 4, the throughput achieved by node 4 over the initial three seconds is still much less than that achieved by nodes 2 and 3. This motivates the need for devising more effective ways to perturb the TDM schedules so as to quickly respond to traffic demands, and to maintain some notion of fairness. Before we describe further refinements towards this end, however, we present a simple model that offers analytical insights into the average throughput performance.

B. Approximate Protocol Modeling

For our analytical model, we assume that each node has a pseudowired link to each neighbor. In addition, we make the following simplifying approximations that decouple the complex interactions among neighbor nodes or links, and allow us to capture the essential tradeoffs in a compact manner:

- We focus attention on protocol dynamics at a single "typical" node, as an approximate representation of interior nodes in a large network. We also assume that all time-slots in a frame evolve independently, and develop a Markov model for the state of a given slot over multiple frames.
- Each node maintains state information on each of its outgoing and incoming links for each slot. Outgoing links can be in one of the following states: "Transmit" (T), "Idle" (I) and "Blocked" (B), where respectively the link is actively transmitting, idle or blocked in the corresponding time-slot. The MDMAC protocol allows only links in the "Idle" state in a given slot to contend for a reservation. For incoming links, this simplifies to only two states: "Receive" and "Idle". In addition, we introduce the "Unavailable" (U) state, indicating that the link cannot contend for that slot because some other link for that node is in "Transmit" or "Receive" state.
- We approximate the state of the links of a node as independent of the state of the links for all other nodes. Clearly, this is not strictly true: a "Transmit" state at node A sending to node B automatically implies that node B is in "Receive" state for that slot. However, this decoupling approximation provides a convenient scenario for analysis.
- The schedule activated on each slot is chosen randomly and independently from the other slots. This is reasonable for a saturated network where all nodes always have traffic to transmit on all of their links.
- We approximate the states of different links for a given node (in a given time-slot) as independent.

The state diagram for an outgoing link under this model is illustrated in Fig. 7. We denote the steady state probabilities of state s as P_s, where $s \in \{T, I, B, U\}$, and the transition probability from s_1 to s_2 as $P_{s_1 s_2}$. Let N denote the number of neighbors for a "typical" node. We now introduce a new tunable protocol parameter, corresponding to *nonpersistent*

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

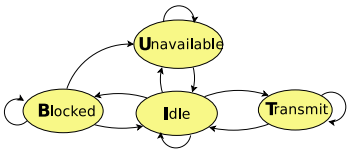


Fig. 7. MDMAC model: state diagram for an outgoing link.

contention: the *listening probability* p_l which is the probability that a node decides not to contend for transmission on any of its outgoing links. Thus, the probability that a node with all links idle chooses to transmit over a given link (each link is chosen with equal probability) is p_{tx}/N, where $p_{tx} = 1 - p_l$. The steady-state probabilities for any state s satisfy:

$$P_s = \sum_{s'} P_{s's} P_{s'} \tag{1}$$

Assuming that the neighbors of the typical nodes are themselves typical, the transition probabilities for the Markov model in Fig. 7 must satisfy certain consistency conditions. Before developing an iterative procedure for computing these in general, we first consider a simple two-node network.

Modeling a Two-Node Network: For the outgoing link from a node, we observe the following: the Unavailable state means that the incoming link from the neighbor is active, the Idle state means that the incoming link is Blocked or Idle, and the Blocked state means that the incoming link is Blocked or Idle. We now compute the transition probabilities for one of the links. The probabilistic state resets are modeled as follows: T_{slot} and T_{block} are the average transmit/receive slot and blocked slot lifetimes. Therefore,

$$P_{TI} = \frac{1}{T_{slot}},\ P_{UI} = \frac{1}{T_{slot}},\ \text{and}\ P_{BI} = \frac{1}{T_{block}} \tag{2}$$

To compute P_{IT}, note that this state transition requires that the node choose to contend in the slot rather than listen (probability p_{tx}) AND either the receiving node is in Blocked state, OR the receiving node is in Idle state AND chooses to listen (probability p_l). Thus we have:

$$P_{IT} = p_{tx}\left(\frac{P_I}{P_I + P_B} p_l + \frac{P_B}{P_I + P_B}\right) \tag{3}$$

where we used the independence approximation to infer that the conditional probabilities of the incoming link being in the Blocked or Idle states given the reference link's Idle state are proportional to the respective steady-state probabilities. Using similar reasoning, we can evaluate the other probabilities as:

$$P_{IU} = \frac{p_{tx} p_l P_I}{P_I + P_B}, P_{BU} = \frac{p_{tx} P_I}{P_I + P_B}, P_{IB} = \frac{p_{tx}^2 P_I}{P_I + P_B}. \tag{4}$$

Procedure 1 defines an iterative algorithm to compute the steady-state probabilities. Consistency demands that the steady-state probabilities of the Transmit and Unavailable states should be equal for the two-node network because of symmetry, and indeed we find that this is always the case. We now compare the analytical and simulation results for the expected link utilization for successful transmissions for each node. Observe that the total medium utilization in this case would actually be the sum of the transmit link utilization for the two nodes sharing the link capacity. The steady-state probabilities calculated from the Markov chain model are: P_T=P_U= 0.489, P_I= 0.015 and P_B= 0.007. QualNet simulations of the protocol for this setting yields the fraction of the successful transmit and receive state slots as 0.492 each, which demonstrates a close match. The other state fractions are Blocked: 0.013 and Idle: 0.002 - the differences correspond to additional refinements embedded into the actual MDMAC QualNet protocol model (see Section III-D).

Procedure 1 State probabilities computation

1: Initialize $P_I = 1,\ P_T = P_U = P_B = 0$.
2: Use the current values of P_s and (2), (3), (4) to compute the transition probabilities $P_{ss'}$.
3: Use the values for $P_{ss'}$ obtained in Step 2 and (1) along with the normalization condition $\sum_s P_s \equiv 1$ to solve for the state probabilities and update the values of P_s.
4: Return to Step 2 until convergence.

The preceding reasoning can be extended to obtain the state probabilities $\{P_s\}$ of the "typical" link for an arbitrary network, i.e., for nodes with N bi-directional links for arbitrary N. The algebra is a bit more involved. For instance, unlike for the two-node network, a link being in Unavailable state does not automatically mean that the corresponding incoming link is active; any of the other $2N-1$ links from/to the same node can be active. We illustrate the method by deriving an expression for P_{IT}, the probability of transitioning from Idle to Transmit state on the reference link. This requires that the node chooses to contend for transmission (probability p_{tx}) rather than to listen, and that out of the subset of the N outgoing links that are in Idle rather than Blocked state, the reference link is chosen. We already know that the reference link is in Idle state. The probability of any other link being in Idle state is $P_I/(P_I + P_B)$. Thus, the probability of the reference link being chosen to contend is given by

$$p_c = p_{tx} \sum_{k=0}^{N-1} \binom{N-1}{k} \left(\frac{P_B}{P_I + P_B}\right)^m \left(\frac{P_I}{P_I + P_B}\right)^k \frac{1}{k+1},$$

where $m = N - 1 - k$. In addition, for the contention attempt to be successful, the corresponding receiving link on the neighboring node, must be in the Idle or Blocked states, and the neighboring node must choose to listen rather than contend to transmit itself on one of its Idle outgoing links (if there are any Idle links, or else all its outgoing links must be in Blocked state). The corresponding probability (p_r) is $p_r = p_l(P_I + P_B(1 - \left(\frac{P_B}{P_I+P_B}\right)^{N-1})) + P_B \left(\frac{P_B}{P_I+P_B}\right)^{N-1}$. Furthermore, the receiving node must choose the reference link out of all of its own other $N-1$ neighboring nodes (also "typical" nodes) who also happen to be Idle and contending for this slot (probability $P_I p_c$). Using this we finally have

$$P_{IT} = p_c p_r \sum_{k=0}^{N-1} \binom{N-1}{k} \frac{(P_I p_c)^k}{k+1} (1 - P_I p_c)^m, \tag{5}$$

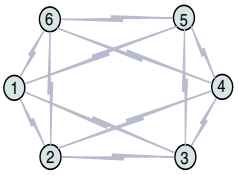


Fig. 8. An example network with 4 neighbors per node.

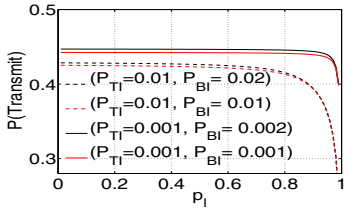


Fig. 9. Medium utilization by node transmissions.

where $m = N - 1 - k$. The expressions for other transition probabilities can be computed using similar reasoning: we omit the details here for lack of space. Procedure 1 can then be used to compute the steady-state probabilities $\{P_s\}$. Note that consistency requires that $P_U = (2N-1)P_T$.

To illustrate the results of the analysis, let us first consider a six-node topology in Fig. 8 where every node has four neighbors with link-saturating flows in each direction. The steady-state medium utilization for successful transmissions from each node is obtained as 0.426. Packet-level simulation of the same scenario yields 0.43. We now employ our analytical model to get insight into the effect of the following parameters on the performance of MDMAC (1) P_{TI}, P_{UI}, and P_{BI}; and (2) $p_l = 1 - p_{tx}$, under saturated traffic conditions and a given network node density. We consider the probabilities P_{TI} and P_{UI} such that the average transmit/receive state lifetime T_{slot} is in the range of 100 to 1000 frames, which correspond to the P_{TI}, P_{UI} values of 0.01 to 0.001. We choose these values based on the criterion that the schedules should be maintained for a time that is one or two orders of magnitude larger than the worst-case packet round trip times for nodes which are multiple hops away from a gateway. In order to understand the effect of tuning the state lifetimes, for each value of P_{TI}, we consider the following Blocked state reset probabilities: $P_{BI} = P_{TI}$ or $P_{BI} = P_{TI}/2$. For the symmetric six-node topology, Fig. 9 shows the expected total medium utilization by a node's transmissions to all its neighbors. We only show a sample of our results to highlight the insights. We find that P_{TI}= 0.001, and P_{TI}= 0.002 yields the best performance among the parameter choices that we have considered. Also, the medium utilization is relatively insensitive to the value of the listening probability p_l (in the range 0.1-0.7). We observe similar trends for medium utilization over a large range of neighbor densities, and therefore set P_{TI}=P_{UI}= 0.001 and P_{BI}= 0.002 for our evaluation (Section V). These relatively large state lifetimes result in less throughput loss due to churn, while (as we shall see) providing enough possibility for rearrangement of schedules to ensure fairness. Although our analytical guidance for choice of P_{TI}, P_{UI} and P_{BI} is for saturated traffic, our simulations indicate that these values are effective for unsaturated multihop mesh traffic as well.

C. Fairness: Explicit State Reset (ESR)

In addition to the preceding mechanisms of probabilistic state reset and nonpersistent contention, we introduce the following strategy to improve the fairness of link bandwidth allocation among neighbors: as long as the fraction of committed transmit or receive slots at a node is less than a threshold (say, $T_{ESR} = 90\%$), its medium access continues as before. However, when this fraction exceeds T_{ESR}, the node successively picks the neighbor that holds the highest share of the committed transmit or receive slots, and resets the state (i.e., considers the slot as free) for a randomly picked slot among the committed slots until the fraction drops below the threshold. Transmit and receive commitments are treated separately while deciding which slot to free up: for many mesh traffic scenarios, the allocation can otherwise be very asymmetric in terms of the transmit and receive slot allocations to a neighbor, which can lead to gross unfairness. The ESR mechanism ensures that when the bandwidth demand at a node starts to approach the total link capacity, the node switches to an "alert" mode to ensure that the bandwidth allocations are not grossly unfair. The node perturbs the TDM schedule by freeing some slots from neighbors that consume the greatest share of the link bandwidth, in order to free up bandwidth for other neighbors. There is no need for this mechanism when the total demand is considerably less than the link bandwidth, but we do need enough "slack" in the schedules such that the probability that a node quickly grabs a free slot is reasonably high. Based on this tradeoff, we have found from our simulations that a 90% threshold works well in practice, in helping competing neighbors obtain a near-fair share of transmit and receive opportunities under heavy traffic.

Fig. 5 shows that the instantaneous flow throughputs with the ESR strategy employed for our toy example (Fig. 2) correspond to a near-fair bandwidth allocation, when nodes 2, 3 and 4 are contending for bandwidth to transmit to node 1. Fig. 6 shows the effect of ESR for multihop flows over our toy network: flows 2→4 and 5→3 start at 1s and 2s, respectively, and are seen to settle into a fair schedule. More sophisticated rules could be applied to decide which slots to free up in order to enforce alternative bandwidth allocation policies (e.g., proportional fairness), but this is beyond our current scope.

D. MDMAC Protocol Design Details

Slot allocation tables: Each node maintains a slot allocation table that contains its current transmit/receive state information. The state information for a particular slot is updated either at the end of the slot, or at the beginning of a new frame: for example, after a successful transmission on a previously idle slot, the slot's state changes from Idle to Transmit, and the associated neighbor information is also stored. For N neighbors, and k slots per frame, the additional memory required to store the state information is $O(Nk)$ units, which is dominated by the worst-case size constraint on the blocked slot list. Thus, the memory and information processing requirements for the protocol do not represent an implementation bottleneck.

Blocked slots: If a node's transmit attempt to a neighbor node fails, the corresponding slot is blocked for future transmit attempts to that neighbor for a geometric number of frames, governed by the probability P_{BI}. However, when there are no unblocked free slots available, blocked slots are picked with a low probability from the pool of free blocked slots for transmit attempts to that neighbor.

Persistence on packet loss on an existing schedule: If a packet transmission fails in the Transmit state, the node changes the state to Tx-Unsure. The node will continue to transmit over the slot, switching back to the Transmit state on a successful packet transmission over the next frames, or freeing up the slot if the packet transmission continues to fail. The state transitions for the receive side are similar: the corresponding states are Receive and Rx-Unsure. The states Tx-Unsure and Rx-Unsure control the level of persistence in the face of failed transmissions *after* the schedule is set to Transmit/Receive: they help to ensure that the schedules are relatively stable even under occasional packet loss. Nodes explicitly communicate to the corresponding neighbor whenever a slot state (Transmit/Receive) is reset to Idle, either because of the probabilistic state reset or the ESR mechanism: this is done via a state-reset bit in the data packet/ACK message headers, or via an ESR message if the transmit packet queue is empty. This communication prevents slot wastage on state reset by avoiding the neighbor's transition to the persistence states described above. *Per-neighbor packet queue backlog-based slot contention*: Each node maintains a per-neighbor output queue, and decides to contend for more transmit slots to a neighbor based on the current queue backlog. For example, if the queue backlog to a neighbor exceeds a threshold T_Q, the node decides to contend for a new slot. In that case, the node randomly picks multiple free/idle slots (e.g., $j{\cdot}p_{tx}$ slots out of total j unblocked free slots) over the frame for transmit attempts to that neighbor. At the beginning of each idle slot, a node first checks if the slot is a candidate for transmission attempts to one or more neighbors, and randomly picks one neighbor (among possibly many) to attempt a transmission. Any slot in the Transmit state is freed if the corresponding neighbor queue is empty.

IV. Medium Utilization with MDMAC

Before delving into detailed simulations that incorporate physical interference, we first examine the properties of the TDM schedules resulting from MDMAC, via the pseudowired model. Consider a directed graph $G = (V, E)$ where V is the set of nodes and the set of edges E represents the directed wireless links between directly communicating neighbor nodes. We define a *feasible schedule* $F \in E$ as a set of links that can be simultaneously activated, i.e., that do not violate the interference constraints (for the pseudowired model this is the half-duplex constraint). We then define a maximal schedule $M \in E$ as a feasible schedule that has the additional property that adding any additional link to the schedule will result in an infeasible schedule. Assume that every node has a link-saturating flow to each of its neighbors. A TDM algorithm is a procedure that maps the set of available slots to the set of feasible schedules. Clearly, any optimal TDM schedule requires that a maximal schedule be used on every slot. While the MDMAC protocol does not guarantee the use of a maximal schedule over every slot, our numerical results show that it does achieve close to optimal schedules with high probability.

Worst-case missed transmit opportunities: We first consider a worst-case estimate of the "missed" transmit opportunities under MDMAC. Specifically, we compare MDMAC against a genie-based procedure in which the genie first looks at the set of successful links activated by MDMAC, and then uses the complete network topology information to schedule transmissions on the largest-cardinality maximal matching of links that is a superset of the links activated by MDMAC. To compute this, we consider a reduced graph R_i of links that can be simultaneously activated with the set of links in the MDMAC schedule. Since MDMAC's schedule over each slot is typically close to a maximal matching, R_i is usually a very small graph, for which the cardinality of the maximum matching can be quickly obtained using a polynomial time maximum matching algorithm as in [23]. With K links in the MDMAC schedule and L additional links found by the genie, the fraction of missed transmit opportunities is $L/(K+L)$. If there was no successful transmission over a slot, we randomly pick a maximal matching for comparison, and set the fraction of missed transmit opportunities to one. For our six node toy network (Fig. 2), we find that on average, MDMAC misses only about 5% of transmit opportunities as compared with the genie-based approach.

Comparison with Greedy Maximal Scheduling (GMS): We also compare MDMAC with a centralized GMS algorithm similar to [24], with the link weights proportional to the fraction of past slots the link was inactive. The latter is a more meaningful weight measure for saturated traffic than the link queue lengths used in [24] as finite-size packet queues would overflow under saturated traffic. The basic idea behind GMS is to incrementally obtain a maximal schedule in the following manner: pick a link l with the maximum weight (breaking ties arbitrarily) among the set of candidate links C (initially set E), remove l and the links that cannot be simultaneously activated with l from the set C, continuing until C is empty. Note that both centralized and distributed versions of GMS have significantly higher control overhead than MDMAC (e.g., see [24]). Our objective here is simply to show that MDMAC provides competitive medium utilization despite its simplicity: for our toy topology (Fig. 2), the expected number of links used per slot for the GMS algorithm turns out to be 2.4, whereas on average, MDMAC uses 2.43 links per slot.

The preceding performance metrics are calculated based on the pseudowired model, which completely ignores signal interference (and hence overestimates performance). We next report on packet-level simulations incorporating a detailed 60 GHz physical layer model that captures topology-dependent (albeit low) interference losses and random packet loss due to noise.

V. Simulation Evaluation

We compare the performance of MDMAC against a directional Slotted ALOHA (DSA) protocol that benefits from the reduced interference by exploiting the highly directional 60 GHz links [6]. DSA serves as a good baseline to understand the potential performance advantages of using memory for medium access coordination. We do not undertake a comparison of MDMAC with the other lower frequency directional MAC protocols, since these protocols are either not applicable because of their underlying assumptions or not

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Parameter	Value
PHY raw data rate, uncoded QPSK link (R)	2 Gbps
PHY overhead (signal acquisition/beamforming) (T_{PHY})	$1\mu s$
Payload + header transmit duration	$(1000+56)*8/R$
SIFS interval	500ns
Total packet transmission+ACK duration	$7.69\mu s$
MDMAC slot duration, frame duration (50 slots)	$8\mu s$, $400\mu s$
$P_{TI} = P_{UI}$, and P_{BI}	$0.001, 0.002$
Queue backlog to trigger a new slot contention	≥ 6 packets
Max. free slots chosen to contend (over a frame, for one neighbor)	0.5*(unblocked free slots), ≤ 10
Prob(a blocked slot picked when no other free slots available)	0.02

TABLE I
SIMULATION PARAMETERS

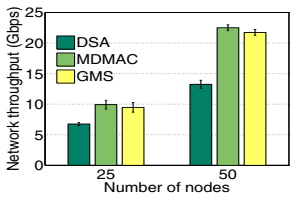


Fig. 10. Aggregate throughput.

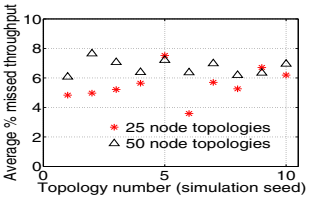


Fig. 11. Missed transmit opportunities.

	25 node topologies	50 node topologies
GMS	0.93	0.93
MDMAC	0.91	0.88
DSA	0.39	0.20

TABLE II
MAC FAIRNESS INDEX (MFI)

designed for the highly directional 60 GHz networks that we consider. For example, most WiFi mesh networking protocols primarily focus on interference management and incur significant control overhead (e.g., RTS-CTS type message exchange, out-of-band tones, directional message broadcasts in each sector, etc.) to avoid interference (see Section I).

In addition to the utilization-related performance metrics described in Section IV, we define a weighted *MAC Fairness Index (MFI)* metric in order to accurately capture the MAC level resource allocation fairness in a saturated network:

$$MFI = \frac{(\sum_{l=1}^{m} y_l/w_l)^2}{m\sum_{l=1}^{m} y_l^2/w_l^2} = \frac{\mu^2(y_l/w_l)}{\mu^2(y_l/w_l)+\sigma^2(y_l/w_l)},\ l \in E$$

where $w_l = 1/max(cs(l), cd(l))$, $cs(l)$ and $cd(l)$ are the number of neighbors of link l's transmit and receive nodes, respectively; m is the total number of links, and μ and σ are the mean and the standard deviation of the weighted link flow throughputs y_l/w_l over all links l. Note that $0 \leq MFI \leq 1$, and for $w_i = 1\ \forall\ i$, MFI equals the Jain's Fairness Index [25].

Simulation Model: We evaluate the performance of MDMAC and DSA via simulations over the QualNet network simulator. We have modified the QualNet physical layer and Antenna modules to model the 60 GHz propagation characteristics and capture our link budget (see Section II). The transmit/receive antenna patterns are obtained from a simple abstraction of the antenna models discussed in Section I. We divide the 360° azimuthal coverage area into 18 sectors. Each sector is covered by a linear array of four highly directive antenna elements (with element pattern approximated by a horn element pattern of directivity 18 dBi). At any instant, the steered antenna patterns and other link budget parameters are used to calculate the signal to interference and noise ratio (SINR) at a node, which then yields the bit error rate (BER) based on the modulation: the packet loss probability evaluated from the BER and the packet size determines successful or failed packet reception. We note that our physical and MAC layer simulation parameters listed in Table I are representative numbers used for illustration; our inferences about MDMAC performance do not depend on these specific parameter choices.

We consider random mesh network topologies of 25 and 50 nodes spread over a 500m x 500m flat terrain. The 50 node topologies are considered to stress-test our protocol design for high-density deployments where interference becomes more significant, and coordination is more challenging. For the mesh topologies with gateway nodes, we run a clustering algorithm over the random topologies to assign gateway nodes in such a way that no mesh node is more than three hops away from its gateway. We simulate the network discovery procedure during the network initialization phase from 0 to 1s. The start times of all the flows are randomly chosen between 1 to 2s. Our simulation results are obtained over 10 simulation runs with a different network topology and traffic for each seed.

Saturated Network Traffic Model: Fig. 10 compares the aggregate network throughput achieved by GMS, MDMAC and DSA. The aggregate network throughput for MDMAC is comparable to the centralized GMS, and much higher than DSA. Table II compares the MAC Fairness Index (MFI) averaged over the different simulation runs. MDMAC's MFI is slightly lower than the MFI for GMS; however, both values are reasonably close to the optimal value of 1. For DSA, the lack of coordination leads to unacceptable link outages for a large fraction of neighbors (20% and 40% on average for the 25 and 50 node topologies, respectively), which translates to DSA's low MFI.

Fig. 11 shows MDMAC's average worst-case missed transmit opportunities: each corresponding to a different simulation experiment, calculated using the procedure described in Section IV. We note that on average, the TDM schedules generated by MDMAC are within 6% and 7% of the corresponding largest cardinality maximal matchings on the network graphs for the 25 and 50 node topologies, respectively. The low missed transmit opportunities demonstrate the efficiency of the distributed TDM schedules that emerge from MDMAC.

Multihop Mesh Traffic Model: This model captures the traffic asymmetries expected in multihop mesh networks. We assign a UDP-CBR flow of rate 200 Mbps from each mesh node to its gateway or/and from a gateway to each node in its cluster, with a probability 0.5. Fig. 12 shows the aggregate network throughput for MDMAC and DSA. MDMAC's implicit transmission coordination results in a significantly higher aggregate throughput than DSA (35% and 52% higher for the 25 and 50 node topologies, respectively). The higher fractional gain for MDMAC over the 50 node topologies demonstrates that MDMAC is effective despite the increased contention and interference resulting from the high node density. We now look at the end-to-end delay and delay jitter for the received packets to obtain insight into the QoS performance. Fig. 13 plots the average packet delay for MDMAC and DSA. Observe that the times taken by the nodes to find slots in MDMAC's schedules over multiple hops turn out to be much lower than the typical Internet traffic delay requirements which are

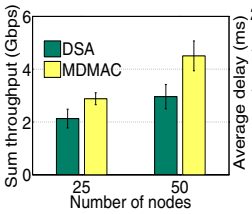


Fig. 12. Throughput.

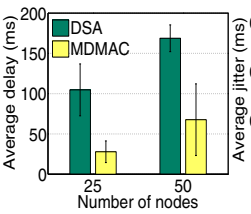


Fig. 13. Total delay.

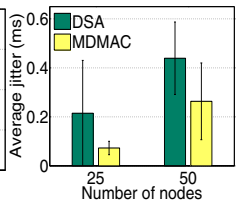


Fig. 14. Delay jitter.

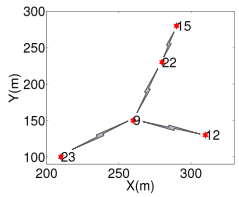


Fig. 15. Portion of a 25 node mesh topology.

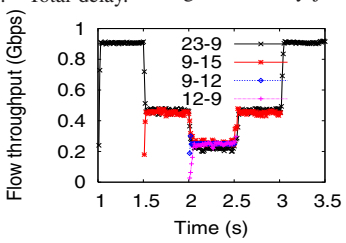


Fig. 16. Instantaneous flow throughputs.

on the order of 100s of milliseconds. MDMAC's TDM-like performance leads to much lower jitter, as seen in Fig. 14.

To illustrate MDMAC's ability to adapt to changes in the traffic pattern, we consider an example with short-lived, high datarate flows, over a portion of a 25 node random topology around a gateway node 9 (see Fig. 15). Fig. 16 shows the instantaneous flow throughputs: flow 23→9 starts at 1s with a throughput of 900Mbps. At 1.5s, flow 9→15 of lifetime 1.5s starts: the schedules quickly adapt such that the two flows share the link bandwidth equally. At 2s, a bidirectional flow (9→12, 12→9) of lifetime 500ms is initiated: MDMAC's explicit state reset mechanism adapts to the changed traffic and readjusts bandwidth allocations for a fair sharing of the total bandwidth. The flow throughputs quickly recalibrate after a flow ends. The results illustrate MDMAC's ability to quickly adapt and maintain link-level fairness.

VI. Conclusions and Future Work

We have shown that 60 GHz mesh networks can operate at high utilization using a completely distributed MAC that exploits the interference reduction, while dealing with the deafness that arises due to the use of highly directional, electronically steerable, beams. The key idea is to obtain TDM-like schedules by using memory about which slots work and which do not, while providing enough random "churn" to allow for quick schedule adaptation. Joint design of routing, scheduling and MAC for highly directional mesh networks, including a comparison of centralized and distributed strategies, remains an important open issue that requires further investigation. Detailed design, implementation, and testing of the envisioned omni-coverage yet highly directional mesh nodes is an important task, which involves design and validation of the antenna array patterns, low-complexity and low-overhead methods for electronic beamsteering, and low-power signal processing techniques for multiGigabit communication. Finally, the promising results in this paper indicate the potential gains from memory and learning-based distributed coordination in more general settings, including omnidirectional networks.

Acknowledgments

This work was supported by the US National Science Foundation under grants CNS-0520335 and CNS-0832154, and by the Institute for Collaborative Biotechnologies under grant DAAD19-03-D-0004 from the US Army Research Office.

References

[1] IBMs 60-GHz Page. [Online]. Available: http://domino.research.ibm.com/comm/research_projects.nsf/pages/mmwave.sixtygig.html
[2] SiBEAM. [Online]. Available: http://www.sibeam.com/
[3] F. Giannetti, M. Luise, and R. Reggiannini, "Mobile and Personal Communications in the 60 GHz Band: A Survey," *Wireless Pers. Commun.*, vol. 10, no. 2, pp. 207–243, July 1999.
[4] S. Alalusi and R. Brodersen, "A 60GHz Phased Array in CMOS," in *Proc. IEEE CICC '06*, Sept. 2006, pp. 393–396.
[5] D. Liu and R. Sirdeshmukh, "A patch array antenna for 60 ghz package applications," in *Proc. IEEE AP-S Symposium '08*, July 2008, pp. 1–4.
[6] R. Mudumbai, S. Singh, and U. Madhow, "Medium Access Control for 60 GHz Outdoor Mesh Networks with Highly Directional Links," in *Proc. IEEE INFOCOM'09, Mini Conference*, Apr. 2009, pp. 2871–2875.
[7] QualNet, v4.1. [Online]. Available: http://www.scalable-networks.com
[8] S. Singh, F. Ziliotto, U. Madhow, E. Belding, and M. Rodwell, "Blockage and Directivity in 60 GHz Wireless Personal Area Networks: From Cross-Layer Model to Multihop MAC Design," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 8, Oct. 2009.
[9] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, and S. Polit, "Ad hoc networking with directional antennas: a complete system solution," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 496–506, March 2005.
[10] Y.-B. Ko, V. Shankarkumar, and N. Vaidya, "Medium access control protocols using directional antennas in ad hoc networks," in *Proc. IEEE INFOCOM '00*, vol. 1, 2000, pp. 13–21.
[11] T. Korakis, G. Jakllari, and L. Tassiulas, "CDR-MAC: A protocol for full exploitation of directional antennas in ad hoc wireless networks," *IEEE Trans. Mob. Comput.*, vol. 7, no. 2, pp. 145–155, Feb. 2008.
[12] R. Choudhury and N. Vaidya, "Deafness: A MAC Problem in Ad Hoc Networks when using Directional Antennas," in *Proc. IEEE ICNP*, 2004.
[13] S. Kulkarni and C. Rosenberg, "DBSMA: A MAC Protocol for Multihop Ad-hoc Networks with Directional Antennas," in *Proc. IEEE PIMRC'05*, vol. 2, Sep 2005, pp. 1371–1377.
[14] Z. Huang, C.-C. Shen, C. Srisathapornphat, and C. Jaikaeo, "A busy-tone based directional MAC protocol for ad hoc networks," *Proc. MILCOM 2002.*, vol. 2, pp. 1233–1238 vol.2, Oct. 2002.
[15] H. Singh and S. Singh, "Smart-ALOHA for multi-hop wireless networks," *Mob. Netw. Appl.*, vol. 10, no. 5, pp. 651–662, 2005.
[16] S. Sanghavi, L. Bui, and R. Srikant, "Distributed link scheduling with constant overhead," in *Proc. ACM SIGMETRICS*, 2007, pp. 313–324.
[17] W. Wang, Y. Wang, X.-Y. Li, W.-Z. Song, and O. Frieder, "Efficient interference-aware TDMA link scheduling for static wireless networks," in *Proc. MobiCom '06*. NY, USA: ACM, 2006, pp. 262–273.
[18] T. Salonidis and L. Tassiulas, "Distributed dynamic scheduling for end-to-end rate guarantees in wireless ad hoc networks," in *Proc. ACM MobiHoc'05*. New York, NY, USA: ACM, 2005, pp. 145–156.
[19] S. Singh, P. Acharya, U. Madhow, and E. Belding, "Sticky CSMA/CA: Implicit synchronization and real-time QoS in mesh networks," *Ad Hoc Netw.*, vol. 5, no. 6, pp. 744–768, 2007.
[20] Y. Yi, G. de Veciana, and S. Shakkottai, "Learning contention patterns and adapting to load/topology changes in a MAC scheduling algorithm," in *Proc. IEEE WiMesh '06*, Reston, VA, USA, 2006.
[21] R. Solis, V. Borkar, and P. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proc. 45th IEEE CDC, San Diego, CA, USA*, 2006.
[22] P. Sommer and R. Wattenhofer, "Gradient Clock Synchronization in Wireless Sensor Networks," in *Proc. ACM/IEEE IPSN '09*, 2009.
[23] H. N. Gabow, "Implementation of algorithms for maximum matching on nonbipartite graphs." Ph.D. dissertation, Stanford, CA, USA, 1974.
[24] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks," in *Proc. ACM MobiHoc '09*. New York, NY, USA: ACM, 2009, pp. 165–174.
[25] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems," Digital Equipment Corp, DEC-TR-301, Tech. Rep., 1984.