Learning from Experience: Efficient Decentralized Scheduling for 60GHz Mesh Networks

Gek Hong Sim[†], Rui Li[‡], Cristina Cano[*], David Malone[§], Paul Patras[‡], and Joerg Widmer[+]

[†]Technical University of Darmstadt, Germany. [‡]The University of Edinburgh, UK. [*]Trinity College Dublin, Ireland.
[§]Maynooth University, Ireland. [+]IMDEA Networks Institute Madrid, Spain.

***Abstract*—Due to the directionality of transmissions in millimeter wave (mm-wave) networks, wireless stations are usually unable to overhear when other stations access the channel. This makes it hard to design efficient distributed beam coordination and scheduling mechanisms. At the same time, centralized schemes only perform well in relatively simple, static scenarios. In practical settings where links exhibit different channel qualities and in the context of relaying or in-band backhauling, centrally coordinating all stations becomes difficult. In this paper, we propose a low complexity, decentralized, learning-based scheduling algorithm for mm-wave networks that handles heterogeneous link rates and packet sizes efficiently. Compared to state-of-the-art slotted channel access for mm-wave networks, the proposed mechanism achieves throughput gains of up to a factor of 8 in single-hop scenarios and end-to-end throughput improvements of up to a factor of 1.6 in multi-hop topologies.**

I. Introduction

In view of the significant mobile data traffic growth currently anticipated [1], millimeter-wave (mm-wave) frequency bands are being explored as a candidate solution to tackle the capacity shortage faced by mobile broadband networks. The very wide (hundreds of MHz to GHz) channels and underutilized spectral resources in these bands open up the possibility of enhancing the capacity of indoor and outdoor wireless deployments and implementing high throughput wireless backhauling. At the same time, mm-wave bands have high path loss, primarily due to carrier-frequency-dependent attenuation and, secondarily, due to oxygen absorption [2]. To overcome this problem, stations employ high gain directional communication, for example through small phased antenna arrays, which allows to confine the emitted energy to narrow beams. This also reduces interference substantially and boosts spatial reuse [3].

Such directional communication, however, introduces *terminal deafness* in the absence of appropriate beam steering and scheduling mechanisms. Therefore medium access solutions previously designed for 802.11 Wireless LANs operating in legacy bands are inappropriate for mm-wave networks. Beam steering has been addressed in the context of 60GHz networks that follow the IEEE 802.11ad standard [4], e.g., through out-of-band angle of arrival estimation [5], to reduce throughput degradation associated with transceivers' beams misalignment.

In addition to identifying the right antenna sector or beam direction, scheduling, i.e., *when* to establish a directional link with the intended receiver, is essential to network performance.

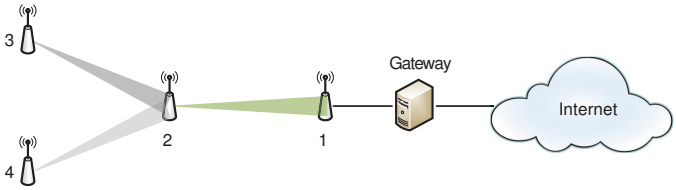


Fig. 1: Simple example of a multi-hop 60GHz network. Station 2 forwards traffic originating at 3 and 4, towards the gateway (node 1).

A simple example scenario is illustrated in Fig. 1, where station 2 forwards traffic from 3 and 4 towards station 1. In the absence of appropriate scheduling, station 2 may loose packets of either station 3 or 4 when it communicates with station 1. While simple centralized single-hop scheduling techniques (e.g., the service period based scheduling mechanisms specified by the IEEE 802.11ad standard [4]) may be sufficient for this basic example, they do not scale to more complex multi-hop relay networks. The reason is that stations need to rapidly and precisely decide which neighbor to beam-steer towards for transmission and reception. Finding a single schedule that suits the entire multi-hop network is a complex problem that typically involves global knowledge and coordination.

Medium access control tailored to 60GHz mesh networks was only considered recently [6], though early efforts fail to capture important practical aspects, including multi-rate operation (due to different link signal-to-noise ratios) and frame aggregation. Precisely, in realistic network settings where links are stable but stations are located at different distances from each other, scheduling over fixed size slots is suboptimal – transmission slots of short duration only allow limited frame aggregation or even require that longer packets are split over multiple slots, whereas long slots are frequently underutilized. In addition, this approach requires alignment of slot boundaries across all stations in the network, which imposes tight synchronization.

In this paper, we tackle the problem of efficient scheduling in multi-hop 60GHz networks through a self-organized approach, Decentralized Learning MAC (DLMAC). DLMAC enables stations to *learn in a decentralized fashion* when to trigger conflict-free directional transmissions, without unnecessarily consuming additional channel resources. With this mechanism, stations operate in an unslotted channel that they divide into cycles of fixed length, comprising a number of micro-

slots. Stations explore randomly chosen micro-slots within an exponentially increasing access window and upon success, the communicating pair reserves the same time interval for directional packet exchanges in subsequent cycles. After that, the transmitter initiates a backward probing procedure to reduce the idle periods in between adjacent allocations (inter-transmission idle time) and improve efficiency. In addition, we propose a micro-slot binary search enhancement, BinDLMAC, which further reduces the inter-transmission idle periods to boost performance.

We demonstrate by means of extensive simulations that our proposal substantially outperforms recent history-based solutions for mm-wave mesh networks [6] in multi-rate and variable packet size scenarios, which makes it particularly suitable for indoor high-speed access networks, in-band back-hauling and multi-hop relaying. The simulation results show that our approach achieves throughput gains of up to a factor of 8 in single-hop networks and end-to-end throughput gains of up to a factor of 1.6 in multi-hop topologies.

The rest of the paper is organized as follows. We present our proposal in Section II and evaluate its performance in Section III. Then, we overview the related work in Section IV and conclude the article with some final remarks in Section V.

II. Decentralized Learning MAC Protocol for 60GHz Networks (DLMAC)

We propose DLMAC, a decentralized learning scheme for scheduling transmissions in 60GHz networks. Stations running DLMAC decide when to transmit based on the outcome of the previous attempts, with the goal of: *i)* finding conflict-free channel allocations, and *ii)* minimizing inter-transmission idle time. In addition, we specify BinDLMAC, which extends DLMAC through a *Micro-slot Binary Search Procedure* (in Section II-E) to further improve channel utilization.

A. Protocol Overview

Our protocol builds upon the recently approved IEEE 802.11ad standard for 60GHz networks [4], which mandates that idle nodes listen in quasi-omnidirectional mode and only switch to directional communication upon a transmission request. This however introduces terminal deafness, i.e. an intended receiver would fail to engage with a transmitter, if already communicating with a different station. To overcome this problem, DLMAC clients independently divide time into cycles of fixed length (schedules) comprised of a number of micro-slots of very small duration and seek to identify non-conflicting sets of micro-slots that can accommodate their transmissions. Stations follow the same cycle length, without requiring to be synchronized and thus the beginning of a cycle can be different for each node.

A node attempting transmission initially picks a set of consecutive micro-slots at random (out of those in the schedule that, to its knowledge, are free) to transmit. If the transmission is successful, the same set of micro-slots will be reserved for future message exchanges in the following schedules at both the transmitter and the receiver. Consequently, both nodes will beam-steer towards each other during the allocated time interval. In case the transmission is unsuccessful, the sender repeats the procedure by choosing at random a different set of micro-slots within an exponentially increasing access window that follows the previous failed transmission attempt.

To improve channel utilization, nodes with established channel allocations probabilistically probe the channel to transmit at an earlier time, with the goal of moving their transmissions closer to other allocations, thereby attempting to cluster packet transmissions together and, thus, prolonging idle intervals to better accommodate future allocations. More specifically, a node will seek to transmit right before its current allocation, such that if the probing is unsuccessful, previously reserved micro-slots can still be used. Figs. 2–3 summarize DLMAC's operation, which we further detail next.

B. Scheduling

In contrast to legacy IEEE 802.11, the lack of carrier sensing due to directional communication prevents nodes from inferring the boundaries of other transmissions, which questions the applicability of slotted channel access schemes to 60GHz networks. Further, mm-wave protocols where a station maintains synchronization and transmissions are confined to fixed length slots (e.g. [6]) perform sub-optimally with varying packet lengths and PHY bit rates – slots are either underutilized or too small to accommodate large payloads.

To address this issue we propose an asynchronous mechanism whereby nodes divide time into schedules that comprise a fixed number of micro-slots, and select a set of these for communication, as depicted in Fig. 2. This approach provides variable-sized allocations to different nodes, allowing DLMAC to adapt better to heterogeneous scenarios with different packet lengths and/or data rates.

We consider the schedule length to be sufficiently long so as to accommodate transmissions in the largest neighborhood and allow for multiple transmissions by the same station in the schedule. Note that a given station may hold multiple allocations within the same schedule, if a suitable set of micro-slots is found for each transmission. In this case, once a node observes that its schedule would not allow for a new station to transmit, it will locally decide to either deallocate one of its transmissions or a reception (by not sending ACKs).

C. Reception Procedure

A node not participating in any communication listens in quasi-omnidirectional mode, so that it can receive requests for communication from its neighbors. If an RTS is received from a particular neighbor during this phase, the node will first assess whether there is enough time to complete the full exchange of CTS, data packet, and ACK, by checking the time left before its next scheduled transmission or reception. In case the full packet exchange can be completed, it will reply with a CTS and upon reception of the data packet, the node will consider this as a scheduled transmission for the next cycle. This is depicted in Fig. 2, where successful allocations in *Schedule 1* are maintained in *Schedule 2*. Right before a

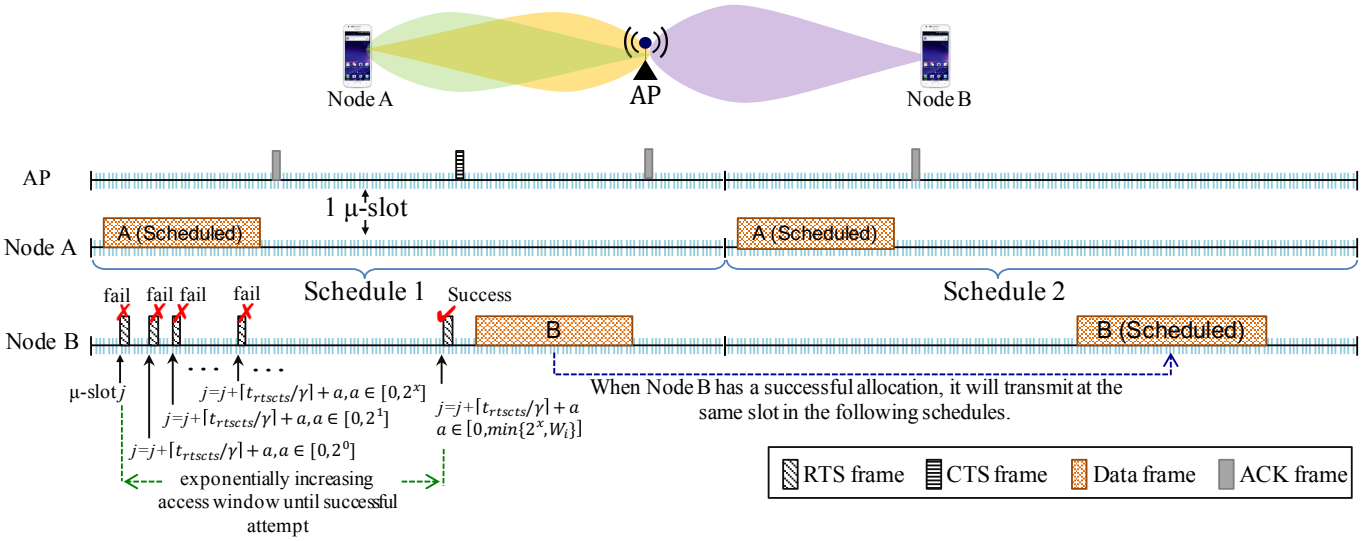


Fig. 2: Two DLMAC stations accessing the channel: schedule, micro-slots and transmission procedure using an exponentially increasing access window upon failed transmissions.

scheduled transmission, nodes involved in the communication switch to directional beams and point these towards each other.

D. Transmission Procedure

We now describe how stations transmit (summarized in Algorithm 1). This involves an initial random channel access followed by packing using RTS probing.

1) Initial Channel Access: A node with a queued packet first sends an RTS in a micro-slot $j \in c(s)$ selected uniformly at random, where $c(s)$ denotes the set of idle micro-slots at schedule $s \in Z^+$. We assume s has sufficient consecutive idle slots to accommodate the transmission[1] (see lines 3–4 in Algorithm 1). If the transmission is successful, i.e., both CTS and ACK are received, the node will consider this attempt as the first successful allocation. The following frames in subsequent schedules are exchanged using the basic access mode (without an RTS/CTS handshake).

If the transmission is unsuccessful, the node infers that the failure may be caused by the receiver being in communication with another station. The node retries in a time slot selected at random from an *exponentially increasing access window* (see Fig. 2). To this end, the station draws a random number a in the range $[0, W_i]$, where i is the number of unsuccessful attempts experienced by that packet and W_i is the corresponding access window. The gap between the transmission attempts will be $j + \lceil t_{\text{rtscts}}/\gamma \rceil + a$ micro slots, where t_{rtscts} = aRTSTime + aSIFSTime + aCTSTimeoutTime, and γ denotes the duration of a micro-slot (lines 13–15). If the attempt is unsuccessful, the station increases the access window and draws randomly a new micro-slot (lines 20–21). This procedure is repeated until a successful transmission occurs.

[1]Recall that a complete transmission comprises the RTS, CTS, data, and ACK frames, which are separated by short inter-frame spacing times (SIFS).

Note, this design speeds up convergence by backing off rather than waiting for the next schedule. While it would be possible to access the channel more aggressively by continuously sending RTSs until transmission is successful, our approach reduces the possibility that transmitter's side lobes may disrupt existing directional links [7].

2) Packing Transmissions via RTS Probing: To reduce the idle periods between transmissions, nodes try to move their allocations closer to other transmissions in the schedule. To this end, once successful, a node starts *RTS probing* (initially with probability $p_{\text{rts}} = 1$) in subsequent schedules. The station sends an RTS in micro-slot $\lceil t_{\text{rtscts}}/\gamma \rceil$ earlier (line 27 in Algorithm 1), allowing enough time for an RTS/CTS exchange before the original transmission is scheduled. If a CTS is received, the transmission is moved back this many microslots, otherwise the existing allocation is retained. This procedure is repeated until no CTS is received, thus packing the transmission closer to earlier allocations in the schedule.

This procedure is illustrated in Fig. 3. In the example shown, *Node B* moves its previous allocation successfully in *Schedule 2* using the RTS probing mechanism. In *Schedule 3*, the probing is unsuccessful and as a result the station keeps its previous successful allocation (lines 36–37 in Algorithm 1).

Initially, RTS probing is limited to a maximum of S/t_{rtscts} times, where S denotes the length of the schedule in seconds (line 31). After this, RTS probing is probabilistically used to address potential gaps caused by nodes leaving the network, while limiting the amount of probing when conditions are more stable. For the initial transmissions, we use $p_{\text{rts}} = 1$ (lines 6 and 17). After a failure in RTS probing (line 37) or when reaching the maximum number of attempts (line 33), a station updates p_{rts} to $\max\{p_{\text{rts}} p_{\text{red}}, p_{\min}\}$, where p_{red} is a reduction factor to p_{rts} to gradually lower the RTS probing probability and $p_{\min}$ is a minimum probing probability to

Algorithm 1 DLMAC - Transmission Procedure

Input:
1: $j \in c(s), s \in Z^+, W_i, t_{\text{rtscts}}, W_{\max} = 128, p_{\text{rts}} = 1.$
Output: j, W_i
2: initialize: $i = 0, W_i = 2^i, m = 0.$
3: choose slot j randomly in $c(s)$
4: access slot j
5: **if** successful **then**
6: $p_{\text{rts}} = 1$
7: **Procedure** RTS probing
8: **else**
9: **Procedure** Exponential access
10: **end if**
11:
12: **procedure** EXPONENTIAL ACCESS
13: increase the access window: $i = i + 1, W_i = 2^i$
14: access range: $a \in [0, \min\{W_i, W_{\max}\}]$
15: access at $j + \lceil t_{\text{rtscts}}/\gamma + a \rceil$
16: **if** successful **then**
17: $p_{\text{rts}} = 1$
18: **Procedure** RTS probing
19: **else**
20: update j; $j = j + \lceil t_{\text{rtscts}}/\gamma + a \rceil$
21: **Procedure** Exponential access
22: **end if**
23: **end procedure**
24:
25: **procedure** RTS PROBING
26: **if** rand$(1) < p_{\text{rts}}$ **then**
27: access at $j - \lceil t_{\text{rtscts}}/\gamma \rceil$
28: **if** successful **then**
29: update j; $j = j - \lceil t_{\text{rtscts}}/\gamma \rceil$
30: **if** $m < S/t_{\text{rtscts}}$ **then**
31: $m = m + 1, p_{\text{rts}} = 1$
32: **else**
33: $m = 0, p_{\text{rts}} = \min\{p_{\text{rts}} p_{\text{red}}, p_{\min}\}$
34: **end if**
35: **else**
36: $k = j - \lceil t_{\text{rtscts}}/\gamma \rceil$
37: $m = 0, p_{\text{rts}} = \min\{p_{\text{rts}} p_{\text{red}}, p_{\min}\}$
38: **Procedure** Micro-slot binary Search
39: **end if**
40: **end if**
41: **Procedure** RTS probing
42: **end procedure**
43:
44: **procedure** MICRO-SLOT BINARY SEARCH [BINDLMAC]
45: access at $k + \lceil (j - k)/2 \rceil$
46: **while** $\lfloor j - k \rfloor > 0$ **do**
47: **if** successful **then**
48: update j; $j = k + \lceil (j - k)/2 \rceil$
49: **else**
50: update k; $k = k + \lceil (j - k)/2 \rceil$
51: **Procedure** Micro-slot binary search
52: **end if**
53: **end while**
54: **end procedure**

ensure the frequency of RTS probing does not become too low. This ensures that when nodes release allocations, arising gaps will be packed quickly. p_{red} and $p_{\min}$ are configurable parameters and we provide suitable values in Section III.

E. Micro-slot Binary Search

Finally, we define a micro-slot binary search mechanism as an extension to DLMAC, referred to as BinDLMAC, to further improve efficiency by minimizing the inter-transmission idle periods. The refinement is motivated by the observation that DLMAC may leave idle time of a duration up to t_{rtscts} between consecutive transmissions. In the *micro-slot binary search* depicted in Fig. 4, a node considers j (its currently allocated micro-slot) and $k = j - \lceil t_{\text{rtscts}}/\gamma \rceil$ (the point at which the last *RTS probing* failed). In the next schedule, the node attempts

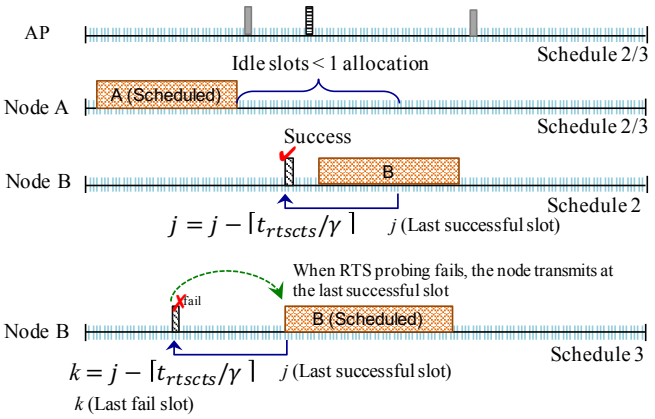


Fig. 3: RTS probing procedure: attempting to move an allocation $\lceil t_{\text{rtscts}}/\gamma \rceil$ micro-slots earlier in the schedule.

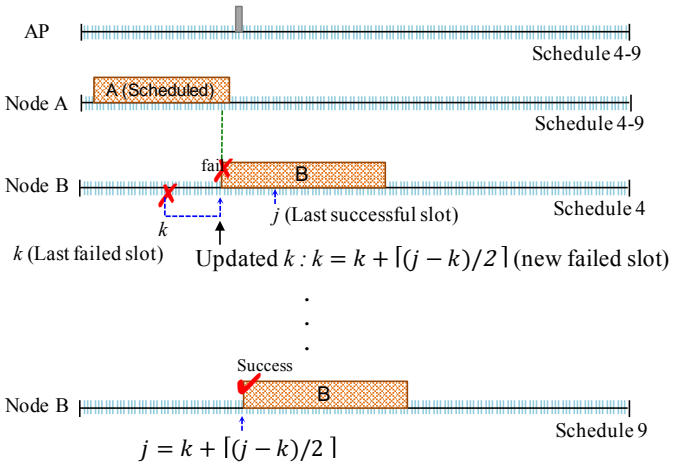


Fig. 4: Micro-slot binary search phase: attempting to transmit at an earlier slot and cluster allocations.

moving its allocated transmission to $k + \lceil t_{\text{rtscts}}/2 \rceil$. Then, upon failure, the node updates k to the new failure point (see line 50 in Algorithm 1) and upon success, it updates j to the new successfully allocated micro-slot (see line 48). The next micro-slot, at which to attempt transmission, will be $k + \lceil (j-k)/2 \rceil$. The search finishes when $\lfloor (j - k) \rfloor = 0$.

III. PERFORMANCE EVALUATION

In what follows, we evaluate the performance of DLMAC and BinDLMAC by conducting extensive simulations over different single- and multi-hop mm-wave network scenarios. We compare our proposals with MDMAC [6], a recent link scheduling protocol for 60GHz networks. Specifically, we measure the aggregate network throughput, when stations operate with the proposed schemes and respectively with different MDMAC versions[2], and transmit frames with varying

[2]By design, MDMAC works with a fixed slot size, optimized for a single payload. For a fair comparison, we examine the protocol's behavior with different slot sizes. We further discuss MDMAC's operation in Section IV.

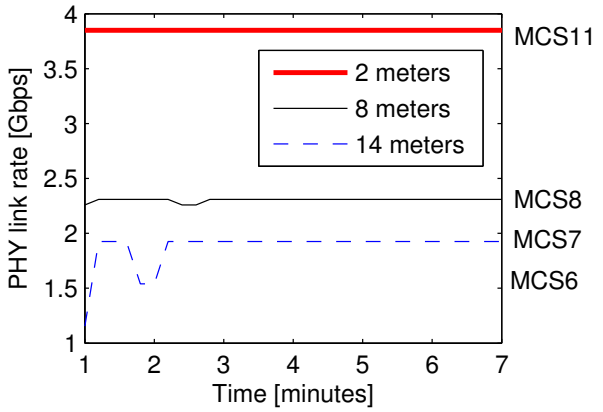


Fig. 5: Experimental testbed results for the MCS selected by a laptop transmitting to a wireless docking station over the 60GHz band for TX–RX distances of 2, 8, and 14 meters.

payload sizes, under both homogeneous and heterogeneous data rates.

For the evaluation, we implement the aforementioned schemes in a Matlab-based, event-driven simulator. We use the signal propagation model given in [2], with the following parameters: oxygen absorption coefficient $\alpha = 0.02$dB/m, carrier wavelength $\lambda = 5$mm, and transmit power $P = 10$dBm. In all the simulations, we configure DLMAC with $p_{\text{red}} = 0.2$ and $p_{\min} = 0.01$.

All protocols comply with the inter-frame and control message durations specified by the IEEE 802.11ad standard, as given in Table I. We assume that the link data rates remain constant during simulation runtime. This assumption is supported by experimental results we obtained in our testbed, using a Dell 6430u laptop and a D5000 wireless docking system equipped with 60GHz transceivers. These experiments confirm that MCS selection is consistent over 7-minutes tests, as illustrated in Fig. 5.

For all simulations, we give averages and 95% confidence intervals for the aggregate throughput, over 50 runs.

TABLE I: IEEE 802.11ad [4] timing parameters.

Parameters	Values
aRTSTime	8.19μs
aCTSTime	8.19μs
aACKTime	6.45μs
aSIFSTime	3μs
aCTSTimeoutTime	15μs

A. Star and random topologies

We first consider two single-hop topologies with ten stations. In the first scenario, nodes transmit to the same AP (star topology), while in the second each station transmits to a randomly selected neighbor (random topology). All stations operate under saturation conditions (i.e., always have packets queued for transmission). Therefore, stations aim to perform multiple allocations within the same schedule. However, a node only attempts to find a new allocation once it successfully completed a packet exchange with an intended receiver. A station will refrain from allocating more transmissions within the same schedule once, to its knowledge, insufficient idle time remains to accommodate other stations.

We investigate scenarios where all stations transmit at a fixed data rate (1.925 Gbps), and respectively where each link operates with a randomly selected bit rate, ranging from 385 Mbps to 4.62 Gbps, corresponding to the 12 single carrier modulation and coding schemes (MCSs) defined by the IEEE 802.11ad standard [4]. We examine the performance of the protocols for different payload sizes $F = \{1.5, 3, 6, 12, 24\}$KB.

1) Star topology, homogeneous data rates: First we evaluate the throughput attained by DLMAC and BinDLMAC under homogeneous link conditions for different payload sizes and compare it to the performance of MDMAC configured with different slot sizes. We depict the results in Fig. 6. In line with our intuition, slotted channel access operating with a fixed slot size only works well when the payload fits the slot size perfectly. More specifically, *(i)* a small slot size leads to packet fragmentation, which may require multiple slots for a single transmission and thus incurs additional overhead (e.g. MDMAC-20μs, $F \geq 3$KB); *(ii)* when the slot size is large, a fraction of the slot remains idle, which reduces protocol efficiency and thus the overall throughput (e.g. MDMAC-160μs, $\forall F$).

In contrast, the aggregate throughput of DLMAC increases monotonically with the payload size and approaches the maximum achievable value in all scenarios. This is due to the fact that DLMAC is inherently more flexible and assigns allocations dynamically.

We note however, that DLMAC attains lower throughput than MDMAC, when the payload exactly matches the slot size. For instance, a $F = 1.5$KB payload requires 19μs for transmission, leaving only 1μs idle time if the slot size is 20μs (MDMAC-20μs). This observation motivates the design

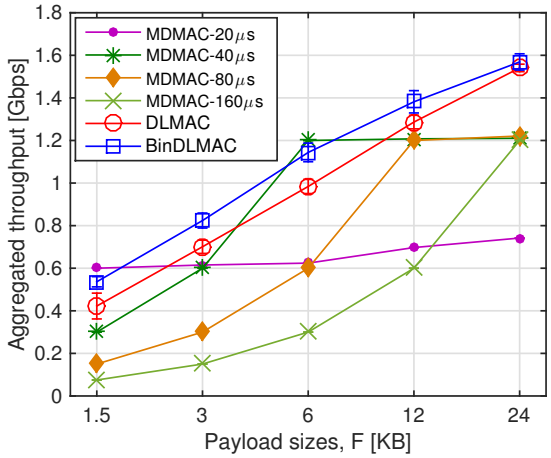
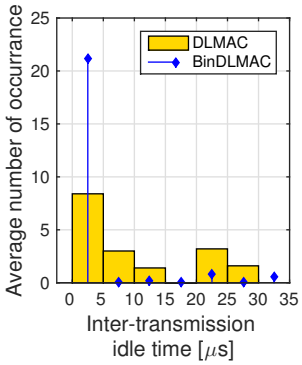


Fig. 6: Throughput comparison between the proposed schemes (DLMAC and BinDLMAC) and slotted channel MDMAC with different slot sizes, for a star topology with $N = 10$ stations transmitting at 1.925Gbps.

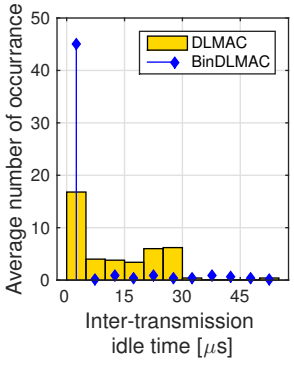
of our BinDLMAC refinement, which seeks to further reduce the inter-transmission idle periods experienced by DLMAC.

Through the micro-slot binary search procedure, BinDLMAC successfully clusters transmissions, which leads to nearly optimal throughput performance. As seen in Fig. 6, by this procedure BinDLMAC achieves up to 25% more throughput than DLMAC and outperforms or performs very close to MDMAC in most settings.

To give further insight into the observed throughput difference, in Fig. 7 we plot the distribution of the inter-transmission idle time when DLMAC and BinDLMAC are used with 1.5 and 6KB payloads. We observe that BinDLMAC does not eliminate large idle times completely, since the probabilistic probing we implement may create inter-cluster gaps. Despite this, BinDLMAC almost triples the number of very short idle intervals (0-5μs), while reducing the number of larger ones, which translates into the throughput gains illustrated in Fig. 6.



(a) Payload size = 1.5KB.



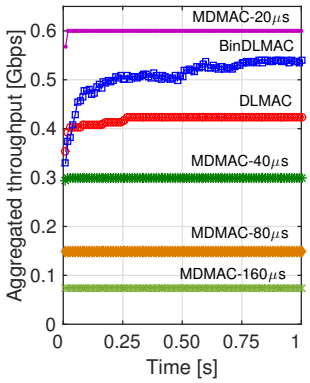
(b) Payload size = 6KB.

Fig. 7: Inter-transmission idle time distributions for DLMAC and BinDLMAC, for 1.5KB (left) and 6KB (right) payloads.

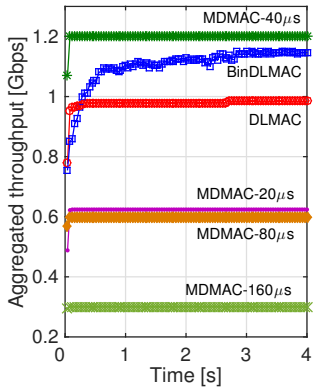
We also examine DLMAC and BinDLMAC's convergence properties in the above scenario. For this purpose, in Fig. 8 we show the evolution of the aggregate throughput for both our approaches and MDMAC. By design, MDMAC stabilizes quickly as all slots are allocated for transmission. In contrast, DLMAC takes slightly longer to converge due to the probing procedure employed to decrease the inter-transmission idle time. Since BinDLMAC further reduces these and improves channel utilization, it requires additional time to converge to a conflict free allocation. Nevertheless, we observe from Fig. 8 that both approaches settle in less than 1 (and respectively 4) second(s) when the payload size is 1.5KB (and respectively 6KB), which we consider acceptable for practical scenarios.

2) Star topology, heterogeneous data rates: Next, we demonstrate that our proposals achieve further performance gains over fixed slot size scheduling mechanisms when links operate with different data rates. We use the same star topology with $N = 10$ transmitters but with link data rates for the different stations chosen randomly from the set of 12 MCSs defined by the standard.

We illustrate the results in Fig. 9, where we plot the aggregate throughput of DLMAC, BinDLMAC, and the different



(a) Payload size = 1.5KB.



(b) Payload size = 6KB.

Fig. 8: Evolution of aggregated throughput in a star topology ($N = 10$ nodes) for DLMAC and BinDLMAC, as well as MDMAC variants for comparison.

MDMAC variants, as we vary the payload size. Observe that in this case, computing an optimal slot size that accommodates a frame perfectly is no longer feasible. As a consequence, all MDMAC variants perform poorly, as the payload size exceeds 1.5KB. In contrast, by employing unslotted channel access and allocating air time adaptively, DLMAC's performance is superior – our approach allocates transmission time individually, depending on both payload size and link rate; this overcomes underutilization of longer slots, as well as the increased overhead associated with short fixed slots. In addition, by reducing the duration of inter-transmission idle time, BinDLMAC further improves network throughput. Specifically, BinDLMAC achieves up to 100% more throughput than MDMAC-20μs and up to 5$\times$ the performance of MDMAC-160μs.

To verify that the observed performance gains are due to MDMAC experiencing high overhead (small slots) or leaving

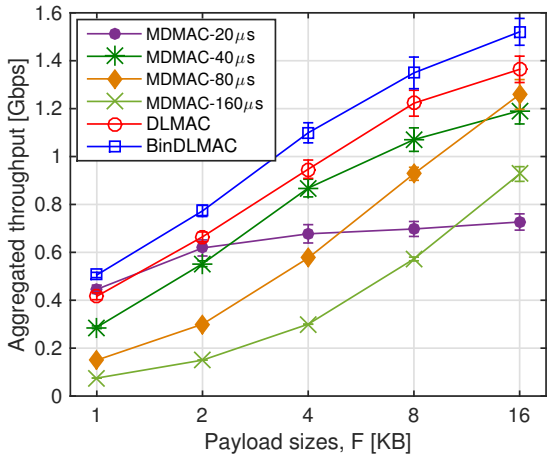


Fig. 9: Throughput comparison between the proposed schemes DLMAC and BinDLMAC as well as the slotted channel MDMAC variants for different payload sizes, for a star topology with $N = 10$ stations with different data rates.

unnecessarily long idle periods within (long) slots, in Fig. 10 we show the percentage of air time for payload transmission (black), overhead (gray), and idle time (white), for both our schemes and the MDMAC variants. Indeed, DLMAC and BinDLMAC consistently utilize a higher fraction of time for payload transmission, while overhead decreases with payload size. In addition, idle time is reduced and protocol efficiency is further enhanced through our micro-slot binary search procedure.

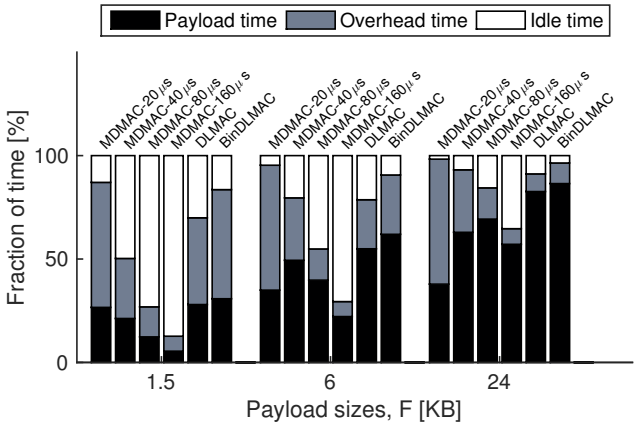


Fig. 10: Fraction of time spent for payload transmission, packet overhead, as well as idle time, for DLMAC, BinDLMAC and the MDMAC variants for a star topology with $N = 10$ stations for different payload sizes and heterogeneous link rates.

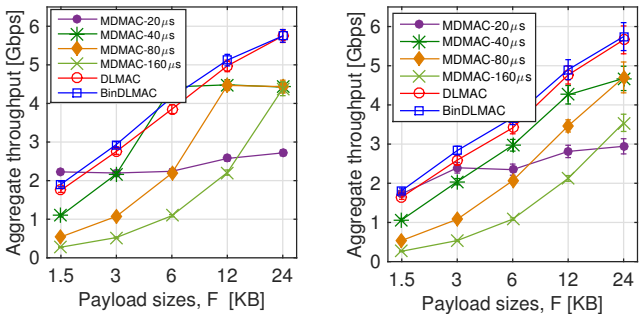
We conclude that no unique slot size exists, such that the performance of slotted access schemes is maximized in all circumstances. By performing adaptive channel time allocation and clustering transmissions, DLMAC and BinDLMAC achieve superior throughput performance and substantially outperform the recently proposed MDMAC scheme.

3) Random topology: Next, we examine a scenario where transmitters do not share the same receiver. More specifically, we consider a 60GHz network with $N = 10$ stations, where each node chooses a destination randomly, in both homogeneous and heterogeneous link rate scenarios. We demonstrate that in such topologies, the aggregate throughput gains of DLMAC and BinDLMAC over MDMAC variants are even higher. To this end, we plot again the network throughput as a function of the payload size when links operate with the same data rate (Fig. 11a) and for randomly chosen data rates among the set of allowed MCSs (Fig. 11b), respectively.

From Fig. 11 we conclude that in the random topologies evaluated, DLMAC achieves up to 8 times higher throughput than MDMAC. The BinDLMAC refinement succeeds in better packing transmissions, which results in further throughput improvements of up to 10% above DLMAC.

B. Multi-hop Topologies

In what follows, we evaluate the performance of the proposed protocols in more complex network scenarios. Specifically, we consider a multi-hop network topology with 20



(a) Homogeneous link rate. (b) Heterogeneous link rate.

Fig. 11: Throughput comparison between the proposed schemes DLMAC and BinDLMAC as well as the MDMAC variants for a random single-hop topology with $N = 10$ stations with data rates of 1.925 Gbps (left) and rates ranging between 385Mbps and 4.62Gbps (right) for different payload sizes.

stations distributed across a 50mx50m area, as depicted in Fig. 13. We compute the corresponding data rate of each link based on the distance between communicating pairs and the propagation model specified by Zhu *et al.* [2], as described above. The antenna sector width of a station is 13° and the maximum distance between two communicating nodes is 23m (which follows the insights gained from our experiments in a real testbed). Although we do not capture the neighbor discovery phase, we note that this could be achieved through beam sweeping [4], or via omnidirectional transmissions in a lower frequency band, as suggested in [5].

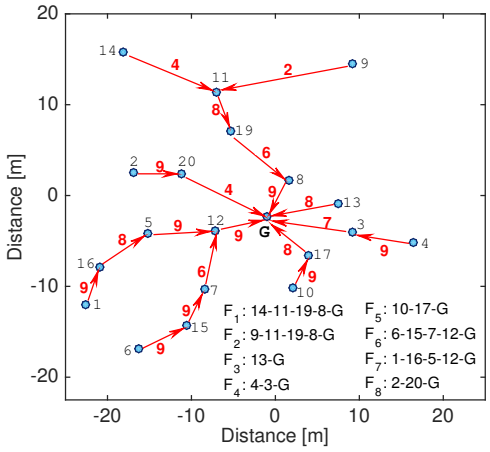
In these simulations, we add further practical considerations, as well as complexity, by assuming flows operate with different payload sizes. Precisely, each flow randomly selects from a set of payloads $F = \{1.5, 3, 6, 12, 24\}$KB. We consider two distinct cases: *(i)* multiple flows originate at different nodes and terminate at the gateway, as indicated by the labels in the bottom right corner of Fig. 12a; and *(ii)* several uplink and downlink flows coexist in the multi-hop topology, as indicated by the arrows and labels depicted in Fig. 12b.

In these scenarios, we measure the end-to-end throughput attained by all flows, and compute the average sum of the individual throughputs over 20 simulation runs, for DLMAC, BinDLMAC, and MDMAC with different slot sizes (between $20 - 160\mu$s). The results of these experiments are shown in Fig. 14, where we observe that also in these multi-hop topologies, as in the single-hop case, DLMAC and BinDLMAC attain substantially higher end-to-end throughput compared to MDMAC.

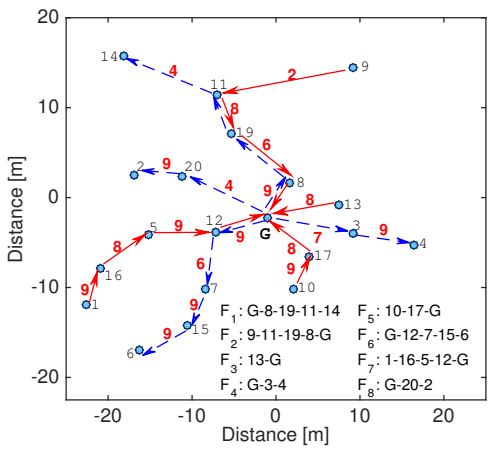
We conclude that, in multi-hop topologies with heterogeneous link rates and frame sizes, by employing unslotted channel access and clustering transmissions, BinDLMAC achieves between 20% and 160% throughput gains over MDMAC.

IV. Related Work

Recent works provide first-hand practical experience of multi-Gbps communications at 60GHz [2], [8] and characterize the highly directional mm-wave wireless links as having some pseudo-wired like characteristics [9]. However, due to



(a) All flows towards gateway.



(b) Flows from and towards gateway.

Mapping of MCS index to data rate as specified in [4].

MCS index	Data rate [Gbps]
2	0.7700
4	1.1550
6	1.5400
7	1.9250
8	2.3100
9	2.5025

Fig. 13: Multi-hop topologies considered for evaluation, with links labeled with their corresponding MCS index (the corresponding data rate is shown in the table on the right) for a pure uplink scenario with all flows terminating at the gateway G (left) and a mixed uplink and downlink scenario (center).

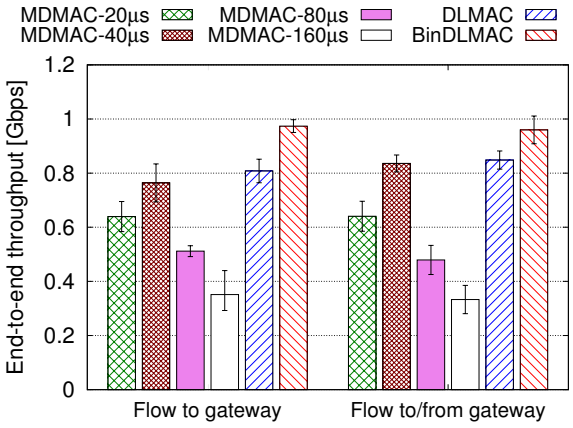


Fig. 14: Comparison of the average sum of end-to-end throughputs attained by the flows shown in Fig. 13, when the stations operate with the proposed schemes and MDMAC variants.

the deafness introduced by the highly directional antenna patterns, carrier sensing is severely reduced and thus legacy CSMA/CA MAC protocols, as for example the traditional 802.11 MAC protocol used for 2.4 and 5GHz bands, are unsuitable for 60GHz links.

Learning-based Scheduling in Wireless Networks: Learning was applied previously in the context of traditional wireless networks to achieve TDMA-like scheduling [10]–[17]. However, carrier sensing enables these earlier schemes to find collision-free slots and determine the schedule length, which is infeasible in mm-wave networks without more complex and high overhead exchange of global information. In contrast, our proposal employs decentralized scheduling for multi-hop 60GHz networks, overcoming terminal deafness without any additional information exchange.

Multi-hop 802.11 Scheduling Solutions: Scheduling methods designed for legacy 802.11 multi-hop networks cannot be applied to 60GHz systems due to fundamental differences that arise with the use of narrow beams. For instance, Choudhury *et al.* propose a directional MAC protocol that employs multi-hop RTS, while CTS, DATA, and ACK are transmitted over a single hop [18]. The approach relies on directional carrier sensing, which cannot be applied on very narrow beams. Laufer *et al.* propose XPRESS, a back-pressure mesh architecture [19], in which a central controller schedules all mesh access points, requiring complex cross-layer information and synchronous operation between the network and link layers. In contrast, our proposed DLMAC does not rely on carrier-sensing, tight synchronization, or complex cross-layer interactions.

60GHz MAC Designs: Given the unique PHY properties of mm-wave bands, the focus in the design of new MAC protocols is shifted from interference management towards overcoming terminal deafness [20]. In single-hop networks, Chandra *et al.* propose to adapt beam widths in mm-wave contention-based access [21] to increase throughput, while legacy 2.4/5 GHz bands are employed in [5] to aid mm-wave technology with beam steering to establish multi-Gbps links. These approaches improve 802.11ad protocol efficiency, but do not address the scheduling problem in the context of deafness.

Chen *et al.* take a first step in this direction and propose a directional cooperative protocol [22], which enables the access point to transform low-SNR single-hop links into multi-hop relayed connections. This solution, however, is centralized and thus has limited scalability in applications such as mm-wave in-band backhauling. To tackle this problem, Singh *et al.* propose MDMAC, a history-based distributed scheduling algorithm for directional 60GHz mesh networks [6]. However, MDMAC addresses scheduling efficiency only to some extent, as the protocol does not capture multi-rate operation and variable packet lengths – it determines a fixed slot size for all transmissions a priori. Further, it involves periodic probabilistic resets of the system state, which potentially degrades performance. In addition, MDMAC's operation requires

synchronization among nodes, which is not trivial in multi-hop topologies. Our proposal tackles these limitations as it does not involve synchronization and is not tied to a fixed slot length. Instead we employ quasi-unslotted access and exploit an effective packing mechanism to improve channel utilization. Consequently, we achieve efficient scheduling in 60GHz networks under steady channel conditions, but with variable link rates.

V. Conclusions

Scheduling solutions for wireless networks mainly target scenarios in which carrier sensing is feasible. However, due to directional transmission used (to cope with critical path loss so as to achieve high throughput) in mm-wave communications, nodes cannot rely on carrier sensing to assess the channel status. In this article, we tackled efficient scheduling for mm-wave networks by applying a decentralized learning approach. In contrast to earlier works, we considered heterogeneous conditions in terms of link data rates and traffic demand across the network. By adopting a quasi-unslotted approach and finding allocations that result in successful transmissions while, at the same time, packing transmissions together to increase efficiency, the proposed protocols achieve 1.6 times the end-to-end throughput of existing approaches in heterogeneous multi-hop topologies, and even higher gains in single-hop scenarios. Moreover, our proposals do not require probabilistically resetting the protocol state to accommodate new transmissions, but instead ensure there is sufficient slack in the schedule to capture network dynamics. The proposed protocols neither require tight synchronization nor information exchange to build and maintain the schedule.

Acknowledgements

This research was funded by the European Research Council grant ERC CoG 617721, the Ramon y Cajal grant from the Spanish Ministry of Economy and Competitiveness RYC-2012-10788, the Madrid Regional Government through the TIGRE5-CM program (S2013/ICE-2919), the LOEWE initiative (Hessen, Germany) within the NICER project, and the University of Edinburgh Development Trust through an Innovative Initiative Grant. The University of Edinburgh is authorised to reproduce and distribute reprints and online copies for their purposes notwithstanding any copyright annotation hereon. In addition, his publication emanated from research supported in part by a grant from Science Foundation Ireland (SFI) and co-funded under the European Regional Development Fund under Grant Number 13/RC/2077.

References

[1] P. Cerwall (ed), "Ericsson mobility report," http://www.ericsson.com/mobility-report, Jun 2015.

[2] Y. Zhu, Z. Zhang, Z. Marzi, C. Nelson, U. Madhow, B. Y. Zhao, and H. Zheng, "Demystifying 60GHz Outdoor Picocells," in *ACM MobiCom*, Sep 2014.

[3] X. Tie, K. Ramachandran, and R. Mahindra, "On 60 GHz Wireless Link Performance in Indoor Environments," in *PAM*, Mar 2012.

[4] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band ," *IEEE 802.11ad Std.*, pp. 1–634, March 2014.

[5] T. Nitsche, A. Flores, E. Knightly, and J. C. Widmer, "Steering with Eyes Closed: mm-Wave Beam Steering without In-Band Measurement," in *IEEE INFOCOM*, May 2015.

[6] S. Singh, R. Mudumbai, and U. Madhow, "Distributed Coordination with Deaf Neighbors: Efficient Medium Access for 60 GHz Mesh Networks," in *IEEE INFOCOM*, Mar 2010.

[7] T. Nitsche, G. Bielsa, I. Tejado, A. Loch, and J. Widmer, "Boon and Bane of 60 GHz Networks: Practical Insights into Beamforming, Interference, and Frame Level Operation," in *ACM CoNEXT*, Dec. 2015.

[8] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez, "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[9] S. Singh, R. Mudumbai, and U. Madhow, "Interference analysis for highly directional 60 GHz mesh networks: The case for rethinking medium access control," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1513–1527, 2011.

[10] J. Lee and J. C. Walrand, "Design and Analysis of an Asynchronous Zero Collision MAC Protocol," *CoRR*, vol. abs/0806.3542, 2008. [Online]. Available: http://arxiv.org/abs/0806.3542

[11] J. Barcelo, B. Bellalta, C. Cano, and M. Oliver, "Learning-BEB: Avoiding Collisions in WLANs," *In Eunice'0 Summer School*, 2008.

[12] M. Fang, D. Malone, K. Duffy, and D. Leith, "Decentralised learning MACs for collision-free access in WLANs," *Wireless Networks*, vol. 19, no. 1, pp. 83–98, 2013.

[13] Y. He, J. Sun, X. Ma, A. V. Vasilakos, R. Yuan, and W. Gong, "Semi-random Backoff: Towards Resource Reservation for Channel Access in Wireless LANs," *IEEE/ACM Transactions on Networking*, vol. 21, no. 1, pp. 204–217, Feb 2013.

[14] C. Cano, D. Malone, B. Bellalta, and J. Barceló, "On the Improvement of Receiver-Initiated MAC Protocols for WSNs by Applying Scheduling," in *IEEE WoWMoM*, Jun 2013.

[15] S. Misra and M. Khatua, "Semi-Distributed Backoff: Collision-Aware Migration from Random to Deterministic Backoff," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 1071–1084, May 2015.

[16] M. F. Tuysuz and H. A. Mantar, "A Beacon-Based Collision-Free Channel Access Scheme for IEEE 802.11 WLANs," *Wireless Personal Communications*, vol. 75, no. 1, pp. 155–177, Aug 2013.

[17] W. Zame, J. Xu, and M. van der Schaar, "Winning the Lottery: Learning Perfect Coordination With Minimal Feedback," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 846–857, Oct 2013.

[18] R. R. Choudhury, X. Yang, R. Ramanathan, and N. H. Vaidya, "On designing MAC protocols for wireless networks using directional antennas," *IEEE Transactions on Mobile Computing*, vol. 5, no. 5, pp. 477–491, 2006.

[19] R. Laufer, T. Salonidis, H. Lundgren, and P. Le Guyadec, "XPRESS: A Cross-layer Backpressure Architecture for Wireless Multi-hop Networks," in *ACM MobiCom*, Sep 2011, pp. 49–60.

[20] R. Mudumbai, S. Singh, and U. Madhow, "Medium access control for 60 GHz outdoor mesh networks with highly directional links," in *IEEE INFOCOM*, Apr 2009.

[21] K. Chandra, R. V. Prasad, I. G. Niemegeers, and A. R. Biswas, "Adaptive beamwidth selection for contention based access periods in millimeter wave WLANs," in *IEEE CCNC*, Jan 2014.

[22] Q. Chen, J. Tang, D. T. C. Wong, X. Peng, and Y. Zhang, "Directional cooperative MAC protocol design and performance analysis for IEEE 802.11 ad WLANs," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 6, pp. 2667–2677, 2013.