Association and Reliability in 802.11ad Networks: An Experimental Study

Mohammed Dahhani, Gentian Jakllari, André-Luc Beylot

IRIT-ENSEEIHT, University of Toulouse, France

{mohammed.dahhani,jakllari,beylot}@enseeiht.fr

Abstract—With the exponential increase in wireless data traffic risking to overwhelm the current networking infrastructures, mmWave networking is seen as on of the most promising paths out of the logjam. IEEE introduced a new standard recently, 802.11ad, that uses mmWave frequencies for delivering multi-Gigabit capacities in a WLAN setting. Such capacities, however, do not come for free: mmWave frequencies have poor propagation characteristics and suffer from blockage, whose consequences on the networking protocols are not fully understood.

In this paper, we aim at in-depth understanding of two key primitives in 802.11ad networks in environments rich with blockage: user association and reliability. Using COTS hardware and a multi-layer tracing tool allowing unprecedented granularity in the analysis of 802.11ad networks, we shed lights on blind spots regarding reliability at both, layer two and layer four.

Index Terms—IEEE 802.11ad, user association, reliability

I. INTRODUCTION

Terrestrial wireless networks operate largely in the microwave frequencies, extending from a few hundred MHz to a few GHz (traditionally referred to as the "beachfront frequencies"). However, these frequencies are starting to suffer from congestion [1] especially since the emergence of the smartphone. Indeed, in its latest Visual Networking Index [2], Cisco reports that mobile data traffic has grown 4,000-fold over the past 10 years and projects that it will increase nearly eightfold by 2021. Fortunately, the millimeter wave frequencies (30-300 GHz) offer several GHz of mostly idle spectrum [3]. In particular, the 57 GHz – 64 GHz spectrum, colloquially referred to as the 60 GHz[1] mmWave band allows unlicensed access across most of the world [1]. This spectrum is not idle by accident – millimeter waves suffer from strong pathloss, atmospheric and rain absorption, poor penetration of obstacles, etc, making it, traditionally, suitable only for short, static, line-of-sight links [4]. Recent advances in semiconductor technology have made it possible to build antenna arrays small enough to fit on a portable devices and capable of forming highly-directional, electronically-steerable energy beams that can overcome the poor propagation characteristics. The technology is mature enough that an amendment to the WiFi family of standards, IEEE 802.11ad, was issued for the 60 GHz [5] supporting Gbps communications. The first 802.11ad products can be now found on high-end laptops and routers [6]. However, the performance of networks based on the new standard is still not well understood. While there is a plethora of works analyzing in detail the link-level performance over mmWave, most are based on non-COTS (commercial off-the-shelf) and/or custom hardware [1], [7]–[12]. Recently, two works provided the first performance studies of networking using fully-compliant 802.11ad COTS products [13], [14]. In [13], the authors provide one of the first evaluations of a WLAN deployment based on 802.11ad and as such the study is aimed at exploring questions related to range, placement and coverage. In [14], the authors use a dense deployment comprising up to 8 nodes to explore the impact of 802.11ad lower layer parameters on upper layer performance.

However, inevitably, there are still unanswered questions regarding the performance of 802.11ad networks. What makes the topic fascinating and worth exploring further is the unique combination of very high data rates, several Gbps, unparalleled in wireless networks, and the unique propagation characteristics of 60 GHz – in particular its vulnerability to blockage. Studies [1], [15], [16] have shown that the simple presence of a person in the line of sight between a client and access point can essentially shut down the link.

In this paper, we aim at a deeper understanding of the networking performance over IEEE 802.11ad COTS devices. Towards this, we focus on two key networking primitives in a WLAN setting: user association and reliability. Given its importance, user association has been subject of many works [17], which usually focus on finding solutions for optimal association and access point (AP) placement. However, we are the first to explore the performance of 802.11ad using COTS hardware under optimal AP placement and discover a major inefficiency in the current implementation of user association. The combination of the inherently lossy wireless channel and the inefficiency of the backoff procedure for very high data rates, led the 802.11ad standard to adopt an enhanced channel access method that involves stations sending a series of packets for every transmission opportunity which are then acknowledged in block by the receiver [5]. However, it is currently unknown how this sophisticated layer-two, window-based mechanism handles link blockage in practice and how it inter-plays with the TCP's window-based transmission mechanism. To address this challenge, we develop a tool that for the first time enables a fine-grained analysis of the reliability mechanism at both layers in 802.11ad networks. Using it on measurement data from our experimental platform led to uncovering blind spots in how blockage in handled by both layers.

---

[1]For the rest of this document, we use the terms mmWave and 60 GHz interchangeably.

In summary, throughout this paper we make the following contributions:

- Using an experimental platform comprising COTS 802.11ad devices (§ II-A), we demonstrate the gravity of blockage in a settings rich with human movements (§ II-B). This is contrary to what was observed in office environments where human movement is transient [13]. With the obvious solution being a multi-AP deployment, we study its potential under the optimal AP placement. This setting allows us to dissect the performance of user association in 802.11ad and uncover a major inefficiency, especially in the mechanism used for allowing a client to disassociate from an access point once the link is blocked so as to associate with an access point with clear line of sight (§ II-C).

- To provide practical insights on how to address the disassociation inefficiency, we perform a careful analysis of the 802.11ad behaviour on COTS hardware when blockage occurs (II-D). We discover that contrary to what happens during normal operation and reported in literature [13], the transmissions of Sector Level Sweep (SLS) frames is significantly accelerated when link quality drops. Back-of-the envelope calculations show that a disassociation mechanism based on SLS frames can accelerate the process by an order of magnitude when compared to the widely accepted beacon-based method.

- We develop a multi-layer tracing tool that, taking as input measurement traces from our experimental platform, allows for a frame-by-frame analysis of the TCP and link layer behaviour in practice (III-B). This unprecedented granularity in the analysis of 802.11ad networks allows us to shed light on blind spots in how both layers handle link blockage (§ III-C). We find that the part of the IEEE 802.11ad protocol regarding its sliding window management is under-specified, leading to different interpretations in practice. Reaching the limits of what is possible on COTS hardware, we resort to trace-driven simulations and identify a throughput-reliability tradeoff in how packet losses on a certain window are managed (§ III-C1). Finally, our measurements show that the TCP if poorly prepared for sudden changes in round-trip timeouts, as those induced by link blockage in mmWave networks, leading to an unnecessary increase in timeouts.

II. User association in 802.11ad networks

In this section, we use COTS hardware to study user association in 802.11ad networks in environments where link blockage is likely.

A. Experimental platform

For our experiments the setup consists multiple TP-Link Talon AD7200 routers [6] that we configure in managed mode or in monitor mode to capture 802.11ad control and data frames.

In order to have more flexibility in our experiments, we ported an OpenWrt adapted by [18] to our TP-Link routers to have more control on the hardware. Typically, using their
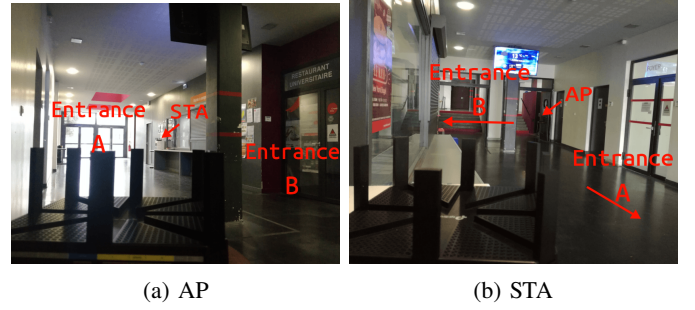

(a) AP   (b) STA

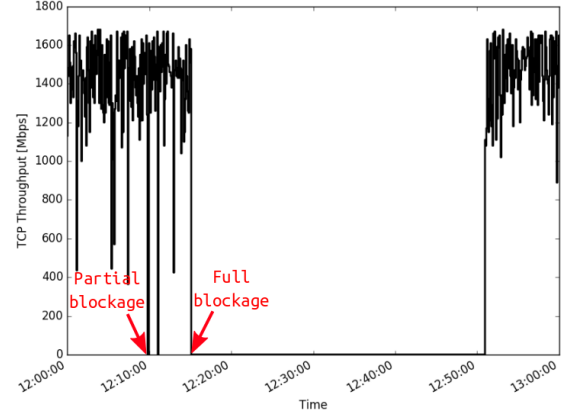Fig. 1: Experimental setup - University restaurant.

Fig. 2: 802.11ad performance in a realistic setting. Minor human movement in the line of sight can have major impact on performance and group movement can completely shut down a link.

firmware patches we were able to access transmitter and receiver Modulation and Coding Scheme (MCS), Signal Quality Indicator (SQI), Signal to Noise Ratio (SNR) and sectors in use by both AP and STA.

B. IEEE 802.11ad link resilience in the wild

Methodology: We perform the following experiment at the entrance of the university restaurant (see Fig. 1): We place a client station and an access point on the opposite sides of the university cafeteria entrance and establish a downlink TCP session. The corridor is $15.3\,\mathrm{m}$ long and $3.2\,\mathrm{m}$ wide and the station and AP are $10\,\mathrm{m}$ away from each-other. The experiment lasted from 12:00 pm to 13:00 pm. Classes finish at 12:00 pm and students start lining up in front of the university cafeteria around 12:05 pm. As indicated in Fig. 1 they enter through entrance A and line up to access the cafeteria through entrance B. By 12:15 pm, the waiting queue builds up and occupies all the space between the entrances A and B.

Results: Between 12:00-12:10 pm the first students start trickling in and there is no queue build up yet. Nevertheless, as they walk across the line of sight between the client and the AP, Fig. 2 shows that their movement has major consequences on link performance. The TCP throughput varies between 1.6 Gbps and 0.4 Gbps. And things only get worse: between
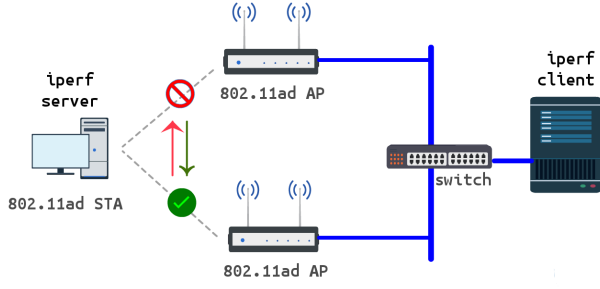
Fig. 3: Experimental Setup - Controlled setting

Fig. 4: Handover effect on link connectivity

12:10 pm and 12:20 pm as students come pouring in, the line outside the cafeteria starts building up. As a result, TCP throughput drops to zero 15% of the time. Past 12:15 pm the queue reaches from entrance A to entrance B thus filling the whole space in-between and throughput drops to zero as the line of sight (LOS) is completely blocked. The link stays broken for the next 30 minutes and only recovers after 12:50 pm, when the number of students joining the queue drastically recedes.

Takeaway: This experiment illustrates a distinct challenge of networking over mmWave: minor human movement between two stations can have major impact on performance and crowded areas can completely shut down a link. This is in contrast with scenarios in which there is little or no human movement, as in [13] and in which a single IEEE 802.11ad can enable ranges of up to $49\,$m.

As a result, for scenarios such as university halls, airports, etc., seamless coverage will require the usage of multiple access points. In a multi-AP infrastructure, users can switch to a better access point when the current link suffers frequent blockage or when the link quality degrades. The problem with this approach is that handover overhead can have serious repercussions on the link [19]–[21]. In an environment where frequent handovers are necessary (e.g. airport, university hall ... *etc.*) the overhead will significantly degrade the performance of time-sensitive applications. The problem is more critical for Gbps links, as is the case of IEEE 802.11ad, where every interruption can translate into significant amounts of data loss. However, handover in IEEE 802.11ad remains poorly understood and is the subject of the next section.

C. Handover under blockage in a multi-AP deployment

As our experimental data in § II-B demonstrated, link blockage can be a significant challenge in IEEE 802.11ad deployments in areas with significant human movement. In a crowded area, blockage of the mmWave link could be mitigated by deploying multiple APs such that whenever the current link suffers blockage, a blockage-free link is always available. Finding such a deployment, formalized as the AP placement problem, has been the subject of several works [22].

In this section, we consider an orthogonal and, to the best of our knowledge, unexplored question: *assuming an optimal placement is already computed, what is the overhead of the handover?*
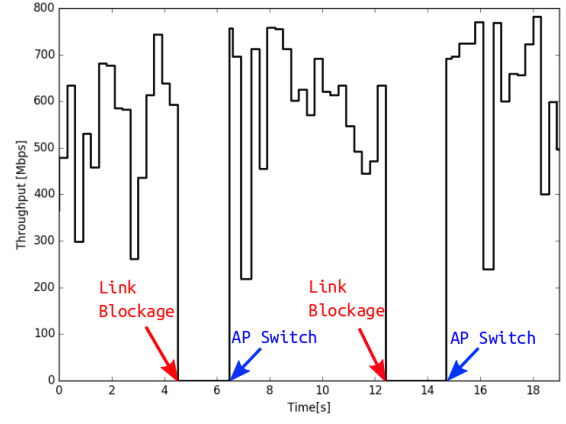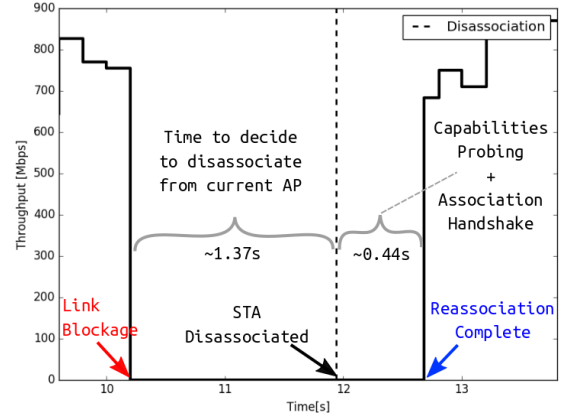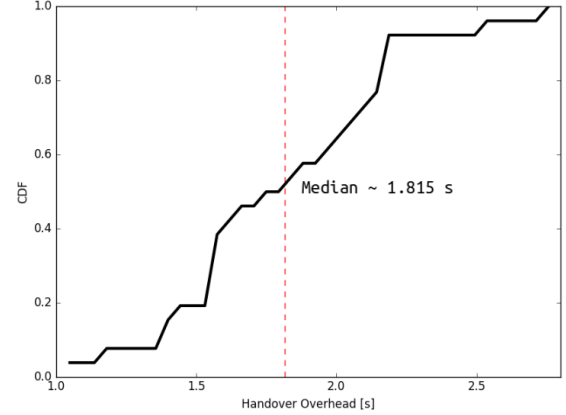

(a) Handover Overhead Dissected

(b) CDF of Handover Overhead

Fig. 5: Handover in 802.11ad network

Methodology: To address this question, first we introduce a definition for the optimal AP placement:

***Definition 1 (Optimal AP placement):** An access point deployment such that for every station there is always at least one access point in range with clear line of sight.*

We apply Definition 1 in setting up a controlled experiment as illustrated in Fig. 3. An iperf client is connected to two 802.11ad access points behind a switch and sends downlink

UDP traffic as fast as possible to an 802.11ad station. The station is placed $50\,\text{cm}$ away from both APs, initially has clear line of sight to both of them. Repeatedly, we carry out the following scenario: We let the station associate with one access point and then block the line of sight to this access point while making sure the line of sight to the other access point is clear. We let the station switch over to the second access point, allow traffic to resume for a few seconds before reversing the position of the blockage, forcing the station to again switch access points.

Results: Figure 4 shows the impact of one full execution of the above scenario. Initially the IEEE 802.11ad station chooses to associate to AP1. After $4.3\,\text{s}$ a human body is inserted in between the station and AP1, while the line of sight to AP2 is kept clear. Since the link is completely blocked, UDP throughput drops to zero. $1.92\,\text{s}$ later the station switches to AP2 and the UDP session resumes. A similar sequence is triggered at t=$12.2\,\text{s}$, blocking the line of sight to AP2 this time. The station switches from AP2 back to AP1 but this time the outage lasts $2.14\,\text{s}$.

To get a more representative evaluation of the handover performance, we repeat the experiment 30 times. Figure 5b shows the CDF of $\Delta_{handover}$, the outage duration due to handover between the two APs. The median outage duration is $1.81\,\text{s}$ and can go up to $2.8\,\text{s}$, very high values for delay-sensitive applications such as VoLTE and 4K video streaming.

Further investigation using a third AP in monitor mode shows that the handover overhead is the sum of two components:

$$\Delta_{handover} = \Delta_{disassociation} + \Delta_{reassocation}$$

where $\Delta_{disassociation}$ is the time between the insertion of blockage and the moment the station receives a disassociation frame[2], and $\Delta_{reassocation}$, the time between the reception of a disassociation frame and the moment the station successfully reassociates with the other access point.

Figure 5a shows a numerical breakdown of the handover process, based on the median values observed during the 30 experiments. The data shows a median value of $0.44\,\text{s}$ for $\Delta_{reassocation}$. It is a high value for a Gigabit link but can be explained by the fact that reassociation involves a station taking time to synchronize with the new AP Beacon Interval (BI), exchange capabilities information, sending an association request and waiting for an acknowledgment from the AP (authentication is disabled). $\Delta_{disassociation}$, the time the AP takes to decide to disassociate its client, with a median of $1.37\,\text{s}$ with a maximum value of $2.23\,\text{s}$, for a link in full outage is, on the other hand, unreasonably high.

Takeaway: Even in a multi-AP deployment where the placement problem is solved optimally (see Definition 1), an IEEE 802.11ad link can still suffers from considerable outage. This is due to the fact that a single handover can take as much as $2.8\,\text{s}$. In a dynamic environment, such as university

halls, airports, etc., where a station may need to continuously switch access points, such handover values can lead to poor performance and user experience.

Our analysis, Fig 5a, shows that a disproportionate amount of time is taken during handover by an access point to realize a link is in outage and decide to disassociate the respective station. Next, we explore how this duration can be reduced.

D. Minimizing handover overhead in 802.11ad networks

In this section, we aim at providing practical insights on how to reduce the handover time in 802.11ad networks. As the experimental data in § II-C showed, a disproportionate amount of the handover time can be attributed to the time it takes for an access point to decide whether the link to a particular station is too poor for transmitting data and disassociate the particular station. Interestingly, the IEEE 802.11ad standard does not define a mechanism for stations to decide when a link is unsuitable for data transmissions and start the disassociation procedure. Our IEEE 802.11ad hardware decides to trigger the disassociation procedure after the link has experienced poor performance for a sustainable amount of time, leading to the high values ($1.37\,\text{s}$ to $2.23\,\text{s}$) observed in § II-C.

An obvious approach for reducing the disassociation time ($\Delta_{disassociation}$), used by regular WiFi for example, is to rely on the beacons the access point transmits periodically. However, beacons are transmitted every BI = $102\,\text{ms}$ and since any decision to disassociate would be based not on a single event but a window of beacon frames, the disassociation time can quickly reach the seconds-level territory. What is needed is a sub-second and reliable mechanism for deciding when a particular link is too poor for data transmissions.

We use a measurement-based approach for discovering a mechanism for sub-second disassociation. In particular, we perform a fine-grained investigation of what happens between the moment blockage is inserted until the moment a station (STA) is disassociated from its access point (AP). For this purpose we use a simple setup with a STA and an AP, and a third node in monitor mode. After the STA is associated to the AP, we block the link until the AP sends disassociation frames. We position the monitor-mode AP next to the STA and use it to capture control traffic.

Our analysis of the control traffic shows that, every time the link is blocked, the received signal strength drops and the AP initiates a Sector Level Sweep (SLS) so as to find a better transmit sector. Our hardware's antenna implements 36 usable sectors for transmission including one quasi-omni directional sector used also for reception. During an SLS period, the AP and STA transmit 36 SLS frames each and conclude the SLS period with a SLS feedback and a SLS feedback acknowledgment.

Figure 6 shows the number of received and missing SLS frames before and after link blockage. During the blockage-free period (Fig. 6a) an SLS exchange happens two times in one second and the STA successfully receives all of the AP SLS frames. However, when the link is blocked (Fig. 6b), an SLS exchange is triggered up to 24 times in a $1/10^{\text{th}}$ of a

---

[2]Control packets are transmitted at the lowest possible rate and can be received even when no data can be delivered.
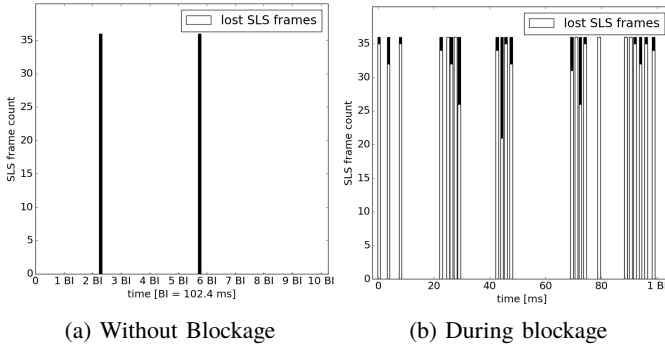
(a) Without Blockage   (b) During blockage

Fig. 6: Correlation between occurrence of blockage and number of lost SLS frames at client-side. During blockage, a large number – 864 (36x24) – of SLS frames is transmitted during a single Beacon Interval (BI), providing ample samples for reliable blockage detection.

	$\Delta_{disasso.}$ [s]	$\Delta_{handover}$ [s]
Hardware	1.37	1.81
Beacon-based	1.02	1.46
SLS-based	< 0.1	< 0.54

TABLE I: Handover schemes comparison.

second! What is more an average of more than 88% of SLS frames are lost. A two-order higher sampling rate than beacons coupled with a high correlation between blockage and SLS frame loss provides a new and far more efficient approach for triggering disassociation.

Table I shows back-of-the envelope calculations of $\Delta_{disassociation}$ and $\Delta_{handover}$ using 2 different detection methods: In the first, AP uses the SLS frame exchange to decide when disassociation should occur and the second is beacon-based. For a fair comparison, our calculations assume that disassociation is triggered after more that 50% losses occur during 10 consecutive periods (be that beacon intervals or SLS periods)

The data shows that while a beacon-based method would be a clear improvement over the approach implemented by our hardware, an SLS-based method would reduce the disassociation time by an order of magnitude when compared to relying on the beacons.

III. Reliability in 802.11ad

In this section, we present a careful, packet-by-packet analysis of how TCP and the IEEE 802.11ad protocol respond to link blockage, a distinct feature of mmWave networking.

A. Primer on 802.11ad link layer transmission reliability

The 802.11ad standard has adopted the Enhanced Distributed Channel Access [5]. A station that wins the contention gains a transmission opportunity and channel access can result in the transmission of a block of data frames spaced by 3 μs. The transmitted block is acknowledged using the block acknowledgment (block ack) mechanism which improves
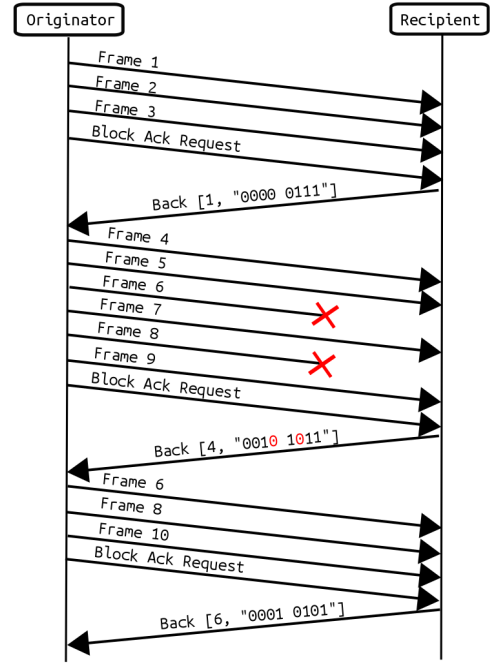

Fig. 7: Link layer retransmission on error

channel efficiency by aggregating several acknowledgments into one acknowledgment frame.

A Directional Multi-Gigabit (DMG) station with data to send is referred to as the originator, and the receiver of that data as the recipient. Two types of block ack mechanisms can be used: *immediate* and *delayed*. When using the immediate block ack mechanism, the recipient must send a block acknowledgment immediately after the reception of a block ack request. This requirement is relaxed using the delayed block ack mechanism where the recipient have the possibility to acknowledge the received frame block later. We limit our study to the immediate block ack mechanism as it is better suited to high-bandwidth and low-latency traffic.

The standard specifies an initialization phase before the block ack mechanism can be used during which recipient and originator negotiate buffer size and block ack format (normal, compressed or multi-traffic identifier format). After initialization, blocks of data frames may be transmitted from the originator to the recipient. The number of frames in the block is limited by the recipient's buffer size. Upon the reception of a block ack request, frames within a block are acknowledged by a block ack control frame which contains a *starting sequence number* subfield and a *bitmap* subfield. The 16 bits of the starting sequence number are used to identify the sequence number of the first frame in the block and the 64 bits -compressed- bitmap is used to acknowledge up to 64 frames, with the first bit referring to the first frame in the block. The recipient acknowledges only frames with a sequence number greater than or equal to the starting sequence number and less than the starting sequence number plus 64. Bits in the bitmap are initialized to 0 and for every received frame, the recipient

sets its corresponding bit in the bitmap to 1.

An illustration of how 802.11ad block ack mechanism is used can be seen in Fig. 7: the first block is successfully acknowledged, then frames 6 and 8 in the second block are lost. The recipient acknowledges only five frames from the second block and sets to zero the 3rd and 5th bits in the block ack's bitmap. Since the starting sequence number is 4, the 3rd and 5th bits in the bitmap refer to frames 6 and 8 respectively. Frames 6 and 8 are retransmitted in the following block along with an additional frame (since the sliding window allows it) and the starting sequence is updated to 6. It is also possible to use a different scheme to retransmit the lost frames; retransmit frames 6 and 8 in a block by itself then wait for the acknowledgment before updating the sliding window. In comparison with the illustrated approach, unless the sliding window's maximum size is reached, reserving a whole transmit opportunity to retransmit lost frames is a waste of medium usage time.

B. Multi-layer tracing

Although [18], [23] proposed a method to access some of the low layer information, we still do not have a method to analyze on a small time scale the dynamics of the link layer. Other important metrics such as link layer usage rate, retransmissions and loss rate are still inaccessible. We did not find any available analysis tool in the literature that would enable us to closely study the performance of the IEEE 802.11ad link layer.

Using iperf3 to generate traffic, netem to control its rate and tcpdump to capture exchanged frames, we developed a trace-based tool to analyze on a frame-by-frame basis the efficiency of the link usage. We use a set of scripts to extract TCP segments and MAC frame sequence numbers. We also extract TCP acknowledgment (ack) sequence numbers and link layer block acknowledgment bitmaps and starting sequence numbers. By comparing sequence space at the transport and link layers we are able to evaluate the cost of link usage by counting link layer retransmissions. And using the MAC/TCP sequence space we are also able to generate time/sequence plots that display both link layer and transport layer activity. Unlike popular packet analyzers, such as Wireshark, we can display in a clear and comprehensive way both link layer and transport layer time-sequence plots. Using tcpdump timestamps we are able to synchronize TCP layer and MAC layers traffic.

Figure 8 shows a TCP sender-side, time-sequence plot generated using a frame capture from a simple two-node experiment. TCP segments (TcpData) are shown by plotting the sequence number of the first byte of a segment. A TCP acknowledgement (TcpAck) is shown by plotting the next sequence number the TCP receiver is expecting. Maximum Segement Size (MSS) can be read from the plot by comparing sequence numbers of two consecutive TCP segements. The plot shows that the first segment is successfully delivered but the second one is lost. Three acknowledgements are received before the TCP sender performs fast retransmit. Besides, the sender does not retransmit segments delivered between
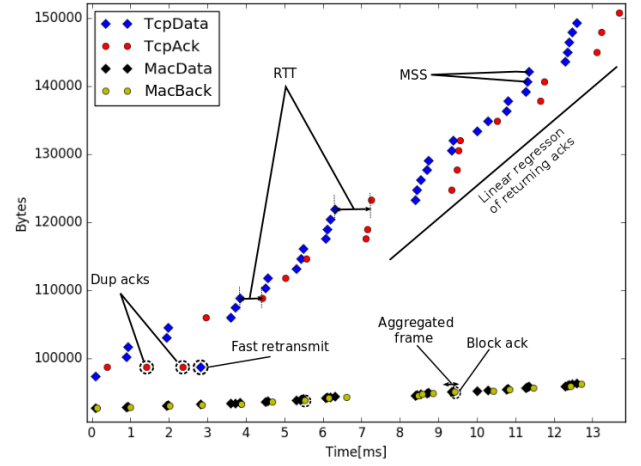

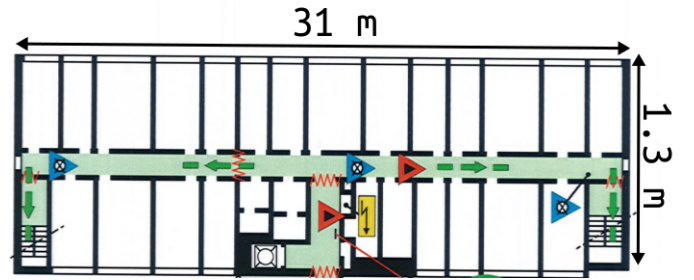Fig. 8: TCP and MAC sender side-time/sequence plot

Fig. 9: Lab corridor floorplan.

the second segment's loss and it's retransmission. In fact, our hardware's TCP version uses selective acknowledgement option (SACK).The TCP receiver informs the sender of non-contiguous blocks of segments that have been received and awaits the reception of the lost segment to fill the gaps in the sequence space. In the time-sequence plot, the third duplicate sack indicates the reception of 4 segments after "the hole" in the sequence space, which triggers fast retransmit. After the 2nd segment is successfully retransmitted, the sender resumes transmission from the 7th segment, since the 3rd up to the 6th segments were selectively acknowledged.

Unlike the TCP segements, the link layer frames are numbered independently of their size. Besides, the 802.11ad link layer does not number frames linearly but using a modulo 4096 [5]. Note that we applied a linear transformation on link layer data and ack frame numbers to fit in the figure. Note also that, in the interest of clarity, we omit from Fig. 8 link layer frames encapsulating TCP acknowledgements.

C. 802.11ad reliability under blockage

In this section, we study the effect of link blockage on transmission reliability in an 802.11ad network. Specifically, we explore how link layer and transport layer retransmission mechanisms behave when the link suffers frequent blockage.

Methodology: We use the experimental setup described in II-A to establish a downlink TCP session between two nodes

in our lab's corridor (Fig. 9). We place a third router next to the sender and set it in monitor mode to capture traffic. We conduct the experiment right before lunch break. The timing was chosen so that people would often block the link. Using the analysis tool introduced in III-B, we generate time-sequence plots in order to observe both the 802.11ad link layer and transport layer reliability mechanisms in action.

Results: First, we observe how the block acknowledgment mechanism at the link layer handles unsuccessful frame transmissions. Figure 10 shows a case where multiple frames in a block were lost. The access point (AP) transmits a 14-frame block and the station (STA) indicates that frames 584, 592 and 595 are missing by setting their respective bits to zero in the first received block acknowledgment bitmap. The AP then retransmits the missing frames which are fully acknowledged by a second block acknowledgment. However, the data shows that the second frame block is composed of retransmit frames only, leading to our first key observation:
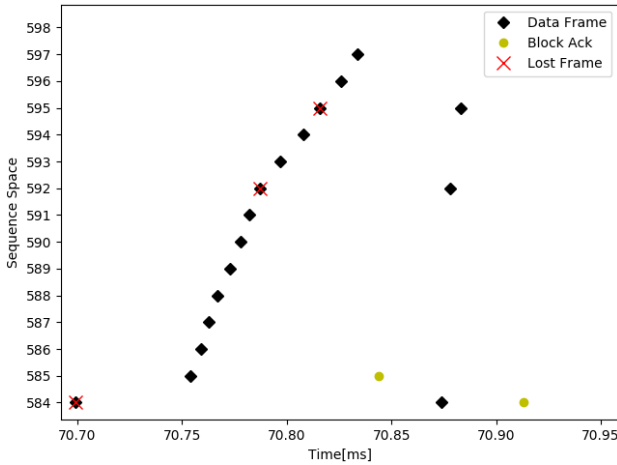

Fig. 10: Sender-side link layer time-sequence plot - Link layer retransmit mechanism

Observation 1: The AP does not send any frames with *fresh* sequence numbers until all losses have been handled.

Our hardware's implementation of flow control using block ack mechanism contrasts with the implementation in [24] in which the link layer is able to aggregate within the same A-MPDU frames from both the retransmission queue and the transmission queue i.e. frames with *fresh* sequence numbers. Both implementations are standard-compliant since the standard's specification of the block acknowledgment operations does not indicate from which queue the MPDU aggregation should be performed. Using an *in-order* or *out-of-order* delivery is implementation dependent.

In-order frame delivery reduces receiver-side frame reordering instances, thus preventing high delays at higher layers. Furthermore, frames transmitted right after a failed transmission could face the same fate in an error-prone channel. However, stopping link layer sliding window advancement every time a frame transmission fails could reduce the number

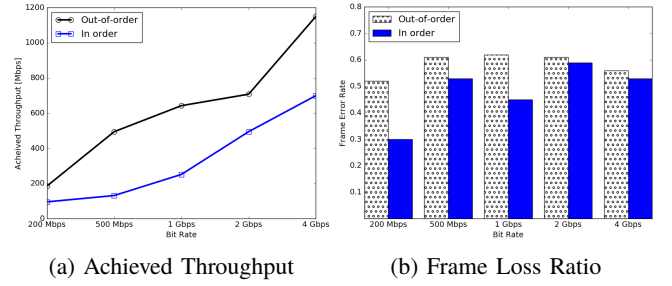(a) Achieved Throughput (b) Frame Loss Ratio

Fig. 11: Throughput-reliability trade-off between in-order and out-of-order retransmission schemes.

of successfully delivered frames with new sequence numbers, i.e. the MAC goodput.

Block transmission was introduced to reduce the wireless medium usage overhead; by acknowledging multiple frames as a block the sender can transmit multiple frames at a time. As such we tried to answer the following question: *Given that a node can have up to 64 frames[3] in transit, is it inefficient to halt the sliding window advancement for every lost frame?*

1) **Window management under losses***:* Since we cannot change the retransmission implementation of the TP-Link hardware, we investigate this question through a trace-driven simulation study that we conduct using an ns-3 802.11ad implementation [24]. We modified the core logic of the code to include an *in-order* retransmission scheme, similar to our hardware's. Using SNR traces we gathered from a real-life environment using our experimental platform, we were able to run trace-based simulations that better reflect realistic scenarios.

The results depicted in Fig. 11 reveal an interesting trade-off: The data shows that, regardless of the used bit-rate, more frames are lost when out-of-order retransmission is used (Fig.11b) while at the same time a higher throughput is obtained (Fig. 11a). This can be explained by the fact that the out-of-order retransmission schemes constantly probes the link layer bandwidth by filling its sliding window as long as there are MPDUs in the transmit and the retransmit queues while respecting the negotiated window size. The conservative in-order approach on the other hand leads to less lost frames since after a frame loss occurs no more frames with a new sequence numbers are transmitted until the lost frame is acknowledged.

Key takeaway: Our analysis combining hardware measurements and trace-driven simulations reveals a throughput-reliability trade-off between in-order and out-of-order reliability in 802.11ad. As the standard does not mandate a particular scheme, we recommend a hybrid scheme capable of dynamically adopting one of the two strategies as a function of the quality of service requirements of the higher layers.

Observation 2 – Blockage-induced TCP timeouts: Figure 12 shows[4] another snapshot of our traces during which

---

[3]Sliding window size is negotiated during the establishment of the block ack agreement

[4]In the interest of clarity, we omit MAC layer frames that encapsulate TCP acks.

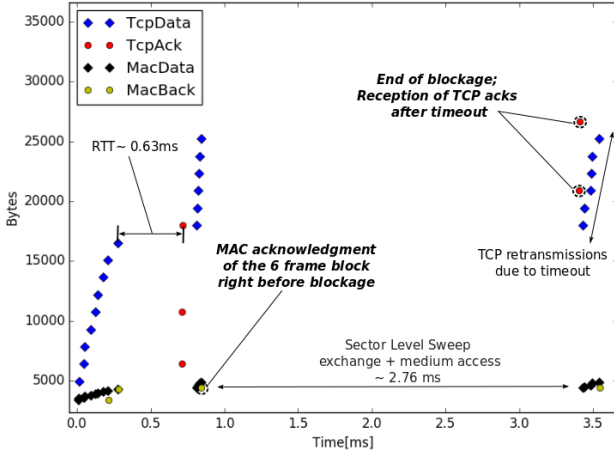a brief blockage event occurred, resulting in an unnecessary TCP timeout.

Fig. 12: Sender-side TCP and MAC time/sequence plot for a blockage-susceptible link

As the time-sequence plot shows, the TCP sender starts with a 9-segment transmit window. The MAC layer aggregates it into two blocks (a 7-frame block and a 2-frame block), which are successfully acknowledge with two block acknowledgments. At the TCP layer, the receiver acknowledges the entire window with 3 TCP acks. One round-trip time (RTT) later (~0.7 ms), the TCP sender updates its transmit window and sends 6 new segments which the MAC layer delivers in one block (aggregation of 6 frames). Five μs later the transmitter receives a MAC block ack with a bitmap confirming the reception of the previous 6 frames. However, the reception of the block ack is immediately followed by a brief blockage event resulting in a received signal strength drop at the transmitter, which triggers a SLS exchange in order to find a better sector. The SLS exchange lasts ~2.15 ms and the link recovers at 3.39 ms. Right after, the sender receives 2 TCP acks confirming the reception of the current 6-segment transmit window. However, the sender does not move its window, retransmitting the current window instead ! This is because during link outage the sender *times out*. Furthermore, after the link recovers, the receiver is the first to win access to the medium. Since the current 6-segment window was successfully delivered right before link outage occurred, the receiver sends 2 TCP acks. To no avail however – the sender has already timed out. As a result, the same current window is retransmitted by the sender.

Specifically, TCP was not able to adjust the retransmission timeout in face of a sudden blockage event and, as a consequence, triggers an unnecessary timeout and retransmission.

Key takeaway: mmWave links are highly susceptible to sudden link outages due to blockage, something the TCP retransmission timeout was not designed to take into account. This specific of the mmWave networks can lead to unnecessary TCP timeouts, well-known to be highly costly to network performance. This challenge, first demonstrated in our mea-

surement study, will need to be taken into account as the community moves forward with the adoption of the mmWave as a backbone of future wireless networks.

IV. RELATED WORK

Practical and robust 60 GHz networking. Related work study the performance of practical 60 GHz networks using both custom hardware platforms [1], [7]–[10] and commercial off-the-shelf devices [11], [12], [25]–[27]. Using experimental hardware poses many problems such as the absence of a standard compliant MAC and higher layers. Also, the bandwidth on such equipment is limited and doesn't make use of the available millimeter-wave spectrum. And finally, most experimental platforms use mechanically steerable horn antennas which inherently add a steering overhead and do not reflect how beam steering is conducted in consumer grade equipment. On the other hand, while consumer-grade devices allow researchers to study performance across multiple layers of the protocol stack and can provide insights into the dynamics at the antenna level, they suffer from a set of limitations: First, they implement pre-802.11ad standard and use proprietary association protocol. Second, they are designed for short-range coverage, LOS and static link use cases, instead of WLAN scenarios which make them unsuitable for efficiently coping with blockage or mobility. Third, access to low layer information is limited. While previous studies [27]–[29], analyze the performance of TCP on mmWave links and emulate multiple APs deployment they lack the in-depth chirurgical approach we use to analyze transport and link layer performances in our study.

Handling 60 GHz link blockage. In the face of blockage, many solutions were proposed and we can group them in the following categories: i) beam realignment [7], [10], [30], [31], ii) beam dilatation [1], [7] iii) reflection based methods [9], [32] iv) out-of-band solutions [33] using Wi-Fi as a fallback interface v) switching to other AP in a dense deployment [28].

AP selection and reassociation. The literature is rich in solutions aiming to diminish the AP handover/reassociation overhead in WiFi networks. Using multiple virtual network interfaces [20], [34], [35] is the most seamless one and does not require any infrastructure modification . The reassociation logic is still applicable in 60GHz networks but the used metric for deciding handover is not e.g. signal strength [36] and location [37], [38]. This is a direct consequence of the characteristics of 60GHz channel: Signal strength drops can indicate the STA is getting out of coverage zone of the current AP or can be an indication of blockage due to environmental dynamics. Node location combined with its orientation [28] can be used to decide when and to which AP reassociate but the used model only considers cases of self-shadowing and loss of alignment while the main need for reassociation in a dense and dynamic environment is frequent blockage. The handover technic we propose does not require location information but only standard-defined MAC control information to detect blockage and to choose an appropriate AP for handover. Not only it does not requires intensive training but it also significantly reduces the handover decision making overhead.

To the best of our knowledge, this is the first proposed solution that leverages MAC control information to make efficient handover decisions in 802.11ad-compliant networks

V. Conclusion Future Work

In this paper we investigated user association and reliability in 802.11ad networks in blockage rich environments. We demonstrated the gravity of blockage in real-life settings and discovered that human movement can have major impact on 802.11ad link performance. By combining hardware measurements, a home-made multi-layer tracing tool and trace-driven simulations we conducted a packet-by-packet analysis of how link blockage affects transmission reliability across the network stack. At the link layer, we identified a throughput-reliability trade-off between *in-order* and *out-of-order* retransmission schemes while at the transport layer we found that due to transient blockage, TCP sometimes fail to adjust retransmission timeout and triggers unnecessary retransmissions. We also analyzed how handover can be used to mitigate link blockage in a multi-AP deployment with optimal AP placement and uncovered an inefficiency in the way disassociation is triggered. We proposed a bacon-based and a SLS-based detection methods that can reduce the outage time due to blockage-induced handover by up to an order of magnitude. A future study is planned for performance analysis of the proposed detection methods.

References

[1] S. Sur, V. Venkateswaran, X. Zhang, and P. Ramanathan, "60 ghz indoor networking through flexible beams: A link-level profiling," in *ACM SIGMETRICS*, 2015.

[2] "Cisco visual networking index: Global mobile data traffic forecast update, 2017-2022," *White Paper*, February 2019.

[3] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: Potentials and challenges," *Proceedings of the IEEE*, vol. 102, no. 3, pp. 366–385, March 2014.

[4] T. S. Rappaport, J. N. Murdock, and F. Gutierrez, "State of the art in 60-ghz integrated circuits and systems for wireless communications," *Proceedings of the IEEE*, vol. 99, no. 8, pp. 1390–1436, Aug 2011.

[5] "IEEE Standards 802.11ad-2012: Enhancements for Very High Throughput in the 60 GHz Band," Dec. 2012.

[6] https://www.tp-link.com/us/products/details/cat-9_AD7200.html.

[7] J. Zhang, X. Zhang, P. Kulkarni, and P. Ramanathan, "Openmili: A 60 ghz software radio with a programmable phased-array antenna: Demo," in *ACM MobiCom*.

[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*, 2013.

[9] G. Zulkuf, H. R. Umar, O. Ertan, and N. Ignas, "Robust 60 ghz indoor connectivity: Is it possible with reflections?" USENIX Association.

[10] H. Hassanieh, O. Abari, M. Rodriguez, M. Abdelghany, D. Katabi, and P. Indyk, "Fast millimeter wave beam alignment," in *ACM SIGCOMM*, 2018, pp. 432–445.

[11] Y. Zhu, Z. Zhang, Z. Marzi, C. Nelson, U. Madhow, B. Y. Zhao, and H. Zheng, "Demystifying 60ghz outdoor picocells," in *ACM MobiCom*, 2014.

[12] S. K. Saha, T. Siddiqui, D. Koutsonikolas, A. Loch, J. Widmer, and R. Sridhar, "A detailed look into power consumption of commodity 60 ghz devices," in *IEEE WoWMoM*, 2017, pp. 1–10.

[13] S. K. Saha, H. Assasa, A. Loch, N. M. Prakash, R. Shyamsunder, S. Aggarwal, D. Steinmetzer, D. Koutsonikolas, J. Widmer, and M. Hollick, "Fast and infuriating: Performance and pitfalls of 60 ghz wlans based on consumer-grade hardware," in *IEEE SECON*, 2018.

[14] H. Assasa, S. Kumar Saha, A. Loch, D. Koutsonikolas, and J. Widmer, "Medium access and transport protocol aspects in practical 802.11 ad networks," in *IEEE WoWMoM*, June 2018, pp. 1–11.

[15] S. Collonge, G. Zaharia, and G. E. Zein, "Influence of the human activity on wide-band characteristics of the 60 ghz indoor radio channel," *IEEE Transactions on Wireless Communications*, 2004.

[16] C. R. Anderson and T. S. Rappaport, "In-building wideband partition loss measurements at 2.5 and 60 ghz," *IEEE Transactions on Wireless Communications*, 2004.

[17] G. Athanasiou, P. C. Weeraddana, C. Fischione, and L. Tassiulas, "Optimizing client association for load balancing and fairness in millimeter-wave wireless networks," *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 836–850, June 2015.

[18] D. Steinmetzer, D. Wegemer, and M. Hollick. (2018) Talon tools: The framework for practical ieee 802.11ad research. [Online]. Available: https://seemoo-lab.github.io/talon-tools/

[19] H. Zhang, Z. Lu, X. Wen, and Z. Hu, "Qoe-based reduction of handover delay for multimedia application in ieee 802.11 networks," *IEEE Communications Letters*, Nov 2015.

[20] A. Zubow, S. Zehl, and A. Wolisz, "BIGAP - seamless handover in high performance enterprise ieee 802.11 networks," in *IEEE/IFIP NOMS '16*.

[21] L. Hao, B. Ng, and Y. Qu, "Self-optimizing scanning parameters for seamless handover in IEEE 802.11 WLAN," in *IEEE LCN*, 2018.

[22] S. Sur and X. Zhang, "Scoping environment for robust 60 ghz link deployment," in *2016 Information Theory and Applications Workshop*.

[23] D. Steinmetzer, D. Wegemer, M. Schulz, J. Widmer, and M. Hollick, "Compressive millimeter-wave sector selection in off-the-shelf ieee 802.11ad devices," in *CoNEXT*, 2017.

[24] H. Assasa. (2016) IEEE 802.11ad mac implementation in network simulator ns-3. [Online]. Available: https://github.com/wigig-tools/ns3-802.11ad/

[25] 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*.

[26] S. K. Saha, V. V. Vira, A. Garg, and D. Koutsonikolas, "Multi-gigabit indoor wlans: Looking beyond 2.4/5 ghz," in *IEEE ICC*. IEEE, 2016.

[27] K. S. Swetank, G. Anuj, and K. Dimitrios, "A first look at TCP performance in indoor IEEE 802.11ad wlans," in *IEEE INFOCOM Workshops*. IEEE, 2015.

[28] T. Wei and X. Zhang, "Pose information assisted 60 ghz networks: Towards seamless coverage and mobility support," in *ACM MobiCom*, 2017.

[29] Y. Zhu, X. Zhou, Z. Zhang, L. Zhou, A. Vahdat, B. Y. Zhao, and H. Zheng, "Cutting the cord: A robust wireless facilities network for data centers," in *ACM MobiCom*, 2014.

[30] A. Loch, H. Assasa, J. Palacios, J. Widmer, H. Suys, and B. Debaillie, "Zero overhead device tracking in 60 ghz wireless networks using multi-lobe beam patterns," in *ACM CoNEXT*, 2017.

[31] X. An, C.-S. Sum, R. V. Prasad, J. Wang, Z. Lan, J. Wang, R. Hekmat, H. Harada, and I. G. Niemegeers, "Beam switching support to resolve link-blockage problem in 60 ghz wpans," *IEEE PIMRC*, pp. 390–394, 2009.

[32] O. Abari, D. Bharadia, A. Duffield, and D. Katabi, "Cutting the cord in virtual reality," in *ACM HotNets*, 2016, pp. 162–168.

[33] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Wifi-assisted 60 ghz wireless networks," in *ACM MobiCom*, 2017, pp. 28–41.

[34] S. Kandula, K. C.-J. Lin, T. Badirkhanli, and D. Katabi, "Fatvap: Aggregating ap backhaul capacity to maximize throughput." USENIX Association, 2008.

[35] V. Bahl, P. Bahl, and R. Chandra, "Multinet: Connecting to multiple ieee 802.11 networks using a single wireless card," in *IEEE INFOCOM*. IEEE, August 2004.

[36] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, and D. Wetherall, "Improved access point selection," in *ACM MobiSys*. ACM, 2006.

[37] P. Venkatadheeraj, K. Hemant, S. Anand, Prabhu, K. Ravi, and C. Malolan, "Location assisted handoffs in dense cellular networks," in *IEEE SECON '14*.

[38] K. Evensen, A. Petlund, H. Riiser, P. Vigmostad, D. Kaspar, C. Griwodz, and P. Halvorsen, "Mobile video streaming using location-based network prediction and transparent handover," in *NOSSDAV '11*.