On Assessing the Performance of IEEE 802.11ad Beamforming Training

Mohammed Dahhani, André-Luc Beylot, Gentian Jakllari

On Assessing the Performance of IEEE 802.11ad Beamforming Training

Mohammed Dahhani, André-Luc Beylot, Gentian Jakllari

Abstract—In the race for Gigabit wireless links, IEEE introduced 802.11ad, an amendment aimed at delivering Gbps capacities in a WLAN setting by leveraging the 60 GHz band. The key innovation of the standard is its beamforming training protocol. Executed periodically at the beginning of every beacon interval, it enables the formation of directional links. To address contention during the uplink part of beamforming training, 802.11ad introduced A-BFT (Association BeamForming Training), an Aloha-inspired, two-level backoff race. While central to the functionality of 802.11ad networks, the performance of A-BFT, however, remains poorly understood.

In this paper, we propose an analytical finite-population model for evaluating the performance of IEEE 802.11ad A-BFT under two channel models: loss-free, and a channel introducing a constant bit error rate. After using an open-source simulator to demonstrate its accuracy, we use our model to assess the performance of A-BFT. We find that a counter-intuitive, quit-easily/be-lazy approach by the stations leads to the best overall beamforming training performance.

Index Terms—IEEE 802.11ad, Random medium access, A-BFT, Analytical performance evaluation

I. INTRODUCTION

IEEE has emerged as a leader in mmWave technology with the introduction of 802.11ad [2], an amendment aimed at offering Gigabit capacities in a WLAN setting. IEEE 802.11ad products operating in the 60 GHz band can be now found on high-end laptops, routers [3] and, most recently, on smartphones [4]. However, if mmWave currently offers several GHz of mostly idle spectrum, it is because it suffers from poor propagation characteristics, making directional antennas essential [5]. As a result, neighbor discovery, the most basic networking primitive, becomes particularly challenging [6]. While with traditional, omni-directional antennas, simply broadcasting a beacon suffices to discover a station in range, with directional communications a catch-22 situation emerges. Discovering neighbors requires communication but communication requires knowing the location of potential neighbors so as to steer the signal beam in the right direction.

IEEE 802.11ad addresses the directional neighbor discovery challenge by introducing a *beamforming training protocol*, executed at the beginning of every beacon interval. In short, it breaks the catch-22 using a two-phase process. First, the access point (AP) transmits a series of control packets in all possible directions (sectors in the standard's language) while stations receive in quasi-omni-directional mode. A station receiving at least one of the control packets is able to identify the best

sector the AP needs to use to reach it. It feeds this information back to the AP in the second phase, during which the roles are reversed. Stations transmit control packets in all sectors while the AP receives in quasi-omni-directional mode. If the AP successfully receives a station's control packets, it can determine: a) the best sector to use to transmit to the particular station and b) the best sector the station needs to transmit to the AP. The process finishes with the AP feeding this information back to the station.

The downlink part of the 802.11ad beamforming training, with only the AP transmitting, is simple protocol-wise. In the uplink part, however, multiple stations may attempt to transmit their control packets to the same access point, leading to contention. To complicate matters further, stations transmit in directional mode, significantly limiting the efficiency of carrier sensing at resolving contention. 802.11ad's answer to this challenge is *Association Beamforming Training (A-BFT)*, a novel, Aloha-like protocol using a two-level backoff race. A-BFT is divided into a fixed number of time slots, with stations allowed to transmit at the beginning of a slot without checking if the channel is busy. Should a collision occur, stations execute a backoff using a fixed contention window equal to the number of slots in A-BFT. If a station fails to transmit its control packets after attempting a maximum number of times, it transitions into idle mode. An idle station is not allowed to transmit and has to enter the idle-state backoff. It selects at random a number between zero and a maximum contention window, set by the standard, and decrements it once per beacon interval. At the end of the backoff, the station can transition back to active.

Fundamental to 802.11ad, the performance of the beamforming training protocol remains poorly understood. To the best of our knowledge, the basic question of how long it will take for a link to be established between a station and an access point – *equivalent to computing the time it takes for a station to successfully transmit its control packets during A-BFT* – remains open. As the first IEEE mmWave technology, recently making its way into consumer products, the primary focus has been on evaluating the overall performance of 802.11ad networks using off-the-shelf hardware and/or simulations [7]–[13]. A few works have considered A-BFT but it has been mostly using simulations [14], [15]. The analytical models introduced in [16], [17] cannot address A-BFT's basic question.

In this work, we aim at introducing an analytical model allowing to assess the performance of A-BFT as a function of its three parameters: 1) the number of slots in A-BFT, N_s; 2) the maximum number of failed attempts in active state, $MaxA$; 3) the idle-state contention window, $MaxI$. While A-BFT

is inspired by Aloha, analyzing it presents new and distinct challenges due to its innovative two-level backoff race and the fact that it is interrupted regularly by the data transmission phase of 802.11ad. In short, we address this challenge by introducing the concept of a virtual slot and a discrete-time Markov chain that accurately models A-BFT.

Our main contributions may be summarized as follows:

- We introduce the first analytical finite-population model that can compute the time it takes for a station to successfully transmit its control packets during A-BFT as a function of its three parameters (Section III). The model takes into account two channel models: error-free and a channel introducing a constant probability of failure. It is simple but nevertheless accurate, modeling exactly what occurs over an A-BFT period and aggregating over k periods.
- We develop a simulator that implements the details of A-BFT as defined in the standard, available to the community as open-source [18], and use it to demonstrate the accuracy of our analytical model (Section IV-A).
- We use our model to assess the performance of A-BFT and find that a counter-intuitive *quit-easily/be-lazy* approach by the stations leads to the best beamforming training performance (Section IV-B). This finding demonstrates the value of an analytical model at optimizing an IEEE 802.11ad network deployment using the parameters provided by the standard.

II. CONTEXT AND PROBLEM STATEMENT

Channel access in IEEE 802.11ad is coordinated using a schedule communicated by the access point (AP). Channel time is divided into Beacon Intervals which in turn are subdivided into *access periods*. As shown in Fig. 1, the first two access periods are dedicated to beamforming training. First the access point, during Beacon Transmission Interval (BTI), transmits control packets in all possible directions while stations receive in quasi-omni-directional mode. The roles are reversed during Association Beamforming Training (A-BFT). The Announcement Transmission Interval (ATI) is then used to exchange management frames and schedule data communication for the DTI period. Finally, the Data Transmission Interval (DTI) is used for data and control frame exchange (Table I summarizes the different acronyms).

Among the mechanisms introduced by IEEE 802.11ad, beamforming training is the most novel and, as a result, the least understood. We focus on it in the following.

A. IEEE 802.11ad beamforming training

The IEEE 802.11ad physical layer defines a set of directional or quasi-omni-directional antenna radiation patterns called *transmit sectors* or *receive sectors*. Beamforming training is the challenge of identifying, from the pre-defined set, the best transmit and receive sectors for establishing a link between two devices.

To address this challenge, 802.11ad introduces the Sector Level Sweep (SLS) [2]. It comprises 3 phases:

1) **Initiator Sector Sweep (ISS)** occurs during BTI. The AP sends a beacon using each of its transmit sectors, including
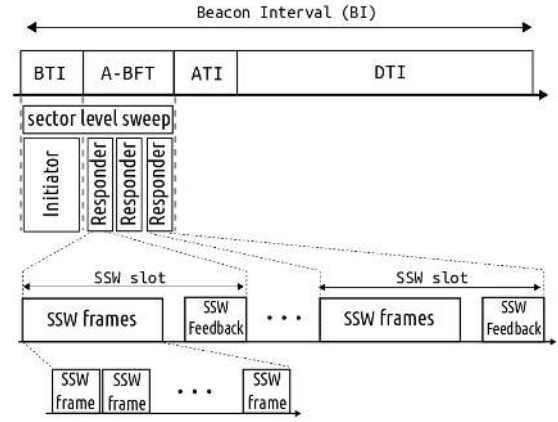

Fig. 1: IEEE 802.11ad beacon interval. A-BFT is composed of several random-access SSW slots. A station gaining access transmits multiple SSW frames, followed by a SSW-feedback by the AP.

in the beacon the sector id, while stations receive in quasi-omni-directional mode. A station receiving one or more directional beacons can identify the best transmit sector the AP should use to reach it.

2) **Responder Sector Sweep (RSS)** occurs during A-BFT. Time is divided into slots (Fig. 1) shared by all the stations through contention. A station gaining access to the medium executes an RSS. It involves sending individual sector-level sweep (SSW) frames on each of its transmit sectors while the AP receives in quasi-omni-directional mode. The SSW frames feed back the transmit sector AP should use to reach the station, identified in ISS, and enable the AP to identify the best transmit sector the station should use to reach it.

3) **Sector Sweep Feedback** also occurs during A-BFT, every time a station completes the transmission of its SSW frames. It is a response from the AP to the station, indicating the best transmit sector the station should use to reach it. An RSS is deemed successful if by the end of the SSW slot the station receives an SSW feedback.

In the next section, we focus on A-BFT, the non-trivial part, protocol-wise, of the beamforming training.

B. Association Beamforming Training (A-BFT)

As shown in Fig. 1, the A-BFT period is divided into a fixed number of Sector-Level Sweep (SSW) slots[1]. Stations compete for the available SSW slots to perform their Responder Sector Sweeps (RSS). If more than one station execute their RSS during the same SSW slot, collisions will occur at the AP, which is receiving in quasi-omni-directional mode. When an RSS attempt fails, the station will have to retry. Two questions emerge regarding the execution of RSS's during A-BFT:

1) How does a station gain access to an SSW slot?
2) What happens when collisions occur?

IEEE 802.11ad addresses these questions using a two-level medium access method. As depicted in Fig. 2, during A-BFT,

[1] For the remainder of the article we will use interchangeably the terms SSW slot and slot, and the terms A-BFT period, A-BFT and period.

TABLE I: BEAMFORMING ACRONYMS

Symbol	Description
BI	Beacon Interval
BTI	Beacon Transmission Interval
A-BFT	Association Beamforming Training
ATI	Annoucement Transmission Interval
DTI	Data Transmission Interval
SSW	Sector Level Sweep
ISS	Initiator Sector Sweep
RSS	Responder Sector Sweep

Fig. 2: The two states of a station.

Fig. 3: Markov Chain model for the Association Beamforming Training (A-BFT). A station can be in active state ($A_1, A_2, ...$) attempting to succeed its RSS, or in Idle state ($I_1, I_2, ...$), attempting to transition to active state.
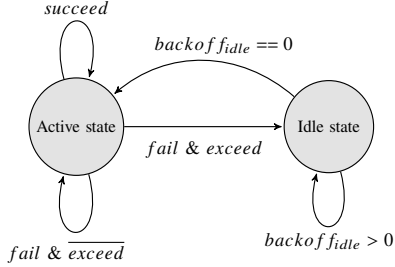
a station can be in one of the two states: *Active*, attempting to execute its RSS using random access; *Idle*, attempting to transition to active state by undergoing a separate random access race. Let N_s be the number of SSW slots, medium access in A-BFT obeys the following rules:

Rule 1 : At the start of an A-BFT period all *active* stations attempt an RSS operation by uniformly choosing a random backoff in $[0, N_s - 1]$.

Rule 2 : At the end of each time slot all *active* stations decrement their backoff counters *even if the medium is busy*.

Rule 3 : If a station's backoff counter is zero at the start of a slot, the station can perform an RSS attempt.

Rule 4 : If a station fails its RSS operation in the current slot, it tries again by choosing a random backoff from the uniform random distribution $[0, N_s - 1]$.

Rule 5 : If the current A-BFT ends before a station's backoff counter reaches zero, the backoff procedure is cancelled and the station tries again in the following period.

Rule 6 : If $MaxA$ successive attempts of the same RSS fail, the station resets its failed attempt count and enters *Idle State*.

Rule 7 : A station in *idle state* stops attempting to transmit an RSS and enters a second backoff race. It chooses another random backoff value from the uniform random distribution $[0, MaxI - 1]$, decremented at the end of each A-BFT period, starting with the following beacon interval.

Rule 8 : A station exits *idle state* and can attempt an RSS in an A-BFT period only if its idle state backoff count is zero at the start of that period.

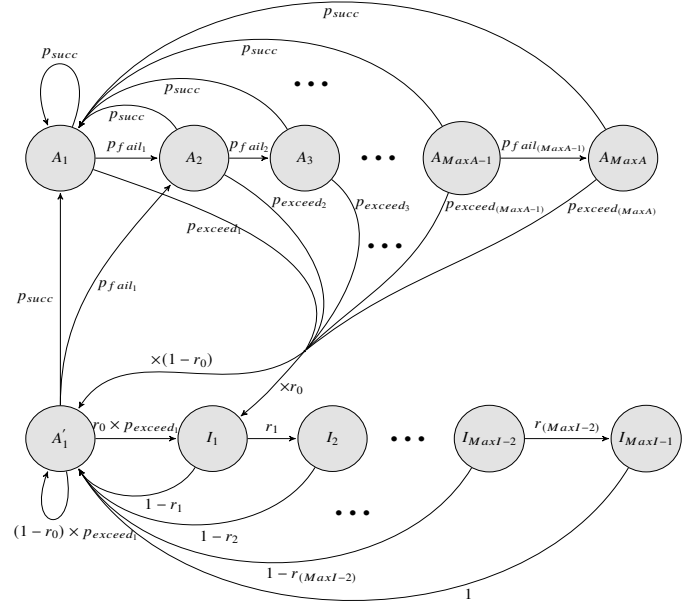*Rule* 9 : A station that succeeds its RSS attempt in an A-BFT period can perform a new RSS in the following period.

C. Problem statement

Given the two-level channel access in A-BFT, the fundamental question becomes:

How long does it take for a station to succeed its RSS and, therefore, complete its beamforming training?

Next, we address this question using an analytical model.

III. AN ANALYTICAL MODEL FOR A-BFT

In this section, we consider an 802.11ad network with N stations and N_s slots per A-BFT period and introduce an analytical model for computing the time it takes for a station to succeed its RSS[2].

A. A Markov chain for A-BFT

Let $MaxA$ be the maximum number of failed attempts, some of which potentially occurring within one period, before a station enters idle state and, $MaxI$, the maximum number of periods a station stays in idle state.

We define $X(t)$ to be the random variable representing the state $(A_1, ..., A_{MaxA}, A'_1, I_1, ..., I_{MaxI-1})$ of a station in A-BFT period t. A discrete time scale is adopted: t and $t + 1$ correspond to the beginning of two consecutive A-BFT periods. A state I_k refers to a period where the station is idle and, states A_k and A'_1, a period where the station is active. Assuming the probability of a successful RSS, p_{succ}, is constant, we can model the process $\{X(t), t \in \mathbb{N}\}$ with the discrete-time Markov chain depicted in Fig. 3. We describe each of its states as follows:

[2]Once associated, future beamforming/tracking can occur in the DTI.

- A station is in state A_1 when it performs a new RSS or after succeeding a previous one. A station is in state A_k if it did not succeed its ongoing RSS for $k-1$ periods since the last time it became active. As long as the station has not reached the maximum number of consecutive failed attempts, $MaxA$, it transitions to state A_{k+1} with a probability of p_{fail_k}.
- If an active station exceeds $MaxA$ consecutive failed attempts $k-1$ periods after the last time it became active, the station becomes idle with probability p_{exceed_k}.
- A station that becomes idle uniformly picks a random number between 0 and $MaxI-1$. It chooses 0 with probability $1-r_0$, becoming active in the next period (state A_1'). Otherwise, it stays idle for at least one period (state I_1), with probability r_0.
- If a station is idle for $k > 0$ consecutive periods (state I_k), it stays idle in the next period with probability r_k (state I_{k+1}), where $r_k = 1 - \frac{1}{MaxI-k}, k \in [0, MaxI-1]$. A station that stayed idle for $MaxI$ consecutive periods becomes active in the following period with probability 1 (state A_1').
- A station transitions to state A_1' after exiting an idle state I_k or in one of the following situations: i) after becoming idle in the same period or multiple periods since the RSS started ii) after becoming active then failing $MaxA$ attempts in the same period. If the station picks 0 in the idle backoff window it ends up in state A_1', resuming the same RSS – as opposed to state A_1 in which the station starts a new one.

B. Computing RSS success time and stationary distribution

The Markov chain in Fig 3 has a finite state space and is ergodic. Therefore the steady-state distribution π exists and is unique. Assuming M is the transition matrix, we find π by numerically solving $\pi = \pi M$.

Let $T_1 = inf\{t \geq 1, X(t+u) = A_1 \mid X(u) = A_1\}$ be the random discrete variable representing the first return time to state A_1. The average number of periods before a station succeeds its RSS corresponds to $\mathbb{E}(T_1)$. State A_1 is by definition recurrent. Therefore, the expected first return time to state A_1 – the RSS success time – can be simply derived using the steady-state distribution π. The average number of periods before a successful transmission is thus $\mathbb{E}(T_1) = 1/\pi_{A_1}$.

The distribution of the access delay T_1 can also be numerically derived using the transition probabilities of the Markov chain:

$$\begin{cases} \mathbb{P}\{T_1 = 1\} = \mathbb{P}\{X(t+1) = A_1 | X(t) = A_1\} = p_{succ} \\ \mathbb{P}\{T_1 = 2\} = \mathbb{P}\{X(t+2) = A_1, X(t+1) \neq A_1 | X(t) = A_1\} \\ \qquad = \sum_{j \neq A_1} \Big(\mathbb{P}\{X(t+2) = A_1 | X(t+1) = j\} \\ \qquad\qquad\qquad \times \mathbb{P}\{X(t+1) = j | X(t) = A_1\}\Big) \\ \mathbb{P}\{T_1 = k\} = \\ \quad \sum_{j_1, j_2, \cdots, j_{k-1} \neq A_1} \Big(\mathbb{P}\{X(t+k) = A_1 | X(t+k-1) = j_{k-1}\} \cdots \\ \qquad \mathbb{P}\{X(t+1) = j_1 | X(t) = A_1\}\Big) \end{cases}$$ (1)

In the following, we compute the one-step transition probabilities of our Markov chain.

TABLE II: MODEL PARAMETERS

Symbol	Description
N	Total number of stations
N_a	Number of active stations
N_s	Number of slots in an period
$MaxA$	Max. number of consecutive failed attempts
$MaxI$	Idle backoff window size
S	Number of successes in a period
π_n	Probability of state $(n_1, .., n_{N_s}, n_\phi)$
$X(t)$	State of station in period t
π	Stationary distribution of $\{X(t), t \in \mathbb{N}\}$
T_1	Time of first return to A_1
τ_{succ}	Success rate in a period
τ_{idle}	Probability a station is idle
L	Number of periods until station becomes idle
$T^{(k)}$	A station's failed attempts after k periods
U_i	A uniform random selection in $[1, N_s]$

C. Computing the transition matrix

Model Parameters: Computing the stationary distribution requires the knowledge of the transition matrix. In this section we will compute the one-step transition probabilities p_{succ}, p_{fail_k} and p_{exceed_k}.

1) **Probability of success**: Let N_a be the number of active stations in a period and $S(N_a)$ (§ III-C3) the discrete random variable representing the number of successful RSS in a period with N_a active stations. The dependence on t is omitted.

We define the success rate in a period as

$$\mathcal{T}_{succ}(N_a) = \frac{\mathbb{E}(S(N_a))}{N_a} \qquad (2)$$

Let $\mathcal{T}_{idle}$ be the probability a station is in idle state. As per rule <u>*Rule* 6</u> (Section II-B), not all N stations contend in an A-BFT period. Some stations can be in idle state. Thus, the number of successful RSSs in a period depends on $N - N_a$, the number of stations in idle state. If we know the probability of a station being in idle state, $\mathcal{T}_{idle}$, and the success rate in a period with N_a active stations, $\mathcal{T}_{succ}(N_a)$, it is possible to compute the success probability of an RSS as:

$$p_{succ} = \sum_{i=1}^{N} \binom{N-1}{i-1} (1 - \mathcal{T}_{idle})^{i-1} \mathcal{T}_{succ}(i) \, \mathcal{T}_{idle}^{N-i} \qquad (3)$$

2) **Failure and entering idle state**: The probability of an active station becoming idle in a given period depends on the number of periods spent in active state (<u>*Rule* 5</u>) and the number of its consecutive failed attempts (<u>*Rule* 6</u>). However, taking into account these dependencies makes the problem of computing the probability of a station becoming idle highly complicated. Thus, we introduce an approximation:

<u>*Approximation*</u> : Our model assumes that the success of a station's RSS in an A-BFT period is independent of the number of periods for which the station has been active. We assume that it only depends on the number of active stations at the beginning of the period and ignore the probability that a station becomes idle before the end of a period. The impact of this approximation is discussed in Section IV-A1.

Computing the conditional probabilities a station transitions into an idle state requires a fine-grain analysis of the dynamic

of RSS attempts in each period. Even by using the above approximation and by conditioning on the failure of a station by the end of the period, finding the probability distribution of the number of RSS attempts is numerically intensive because of the very large state space; a collision in a given slot can be due to a number of previously colliding stations that chose the current slot or to any combination of successive collisions since the start of the period that resulted in a collision in the current slot! For this reason we opted for a lighter approach where we consider RSS attempts in a period as a succession of independent uniform random selections U_i in the interval $[1, N_s]$, such that there was a collision each time.

Let S_j, for $j \in [1, N_s]$, be the sum of the first j uniform random selections and $T^{(1)}$ the random variable representing the number of RSS attempts in one period. $T^{(1)}$ takes its values in $[1, N_s]$. We get:

$$\begin{cases} S_j & = \sum_{i=1}^{j} U_i \\ \mathbb{P}(T^{(1)} = j) & = \mathbb{P}(S_{j+1} > N_s \mid S_j \leq N_s) \end{cases} \quad (4)$$

Let $T^{(k)}$ be the number of failed RSS attempts after k periods. $T^{(k)}$ takes its values in $[1, N_s]$ as well. By performing a convolution product we get the probability distribution of $T^{(k)}$ as follows:

$$\mathbb{P}(T^{(k)} = j) = \sum_{i=1}^{j-1} \mathbb{P}(T^{(k-1)} = i) \, \mathbb{P}(T^{(1)} = (j - i)) \quad (5)$$

Let L be the random variable representing the number of periods until a station reaches its maximum number of failed RSS attempts, $MaxA$. We get:

$$\begin{cases} \mathbb{P}(L = 1) = \mathbb{P}(T^{(1)} \geq MaxA) \\ \mathbb{P}(L = k) = \mathbb{P}(T^{(k)} \geq MaxA \mid T^{(k-1)} < MaxA) \end{cases} \quad (6)$$

Finally, the probability a station fails k periods later but stays active and the probability a station fails k periods later then becomes idle, respectively p_{fail_k} and p_{exceed_k}, can be computed as follows:

$$\begin{cases} p_{fail_k} & = (1 - p_{succ})(1 - \mathbb{P}(L = k)) \\ p_{exceed_k} & = (1 - p_{succ})\mathbb{P}(L = k) \end{cases} \quad (7)$$

Since RSS attempts obey a uniform random process, closed form solutions for equation (6) are straightforward.

Therefore, constructing the transition matrix requires $\mathcal{T}_{succ}$, a station's success rate in a period and p_{succ}, the success probability of a station.

3) **Success rate in an A-BFT period***:* In this section, we focus on what happens within one A-BFT period and propose a theoretical model based on enumerative combinatorics to find the law of $S(N_a)$, the number of successes in a period with N_a active stations.

Finding the success rate τ_{succ} is about identifying how many stations successfully performed RSS by the end of a period. To do so, we introduce the *virtual slot*, a concept we use to model how stations retry an RSS in a period.
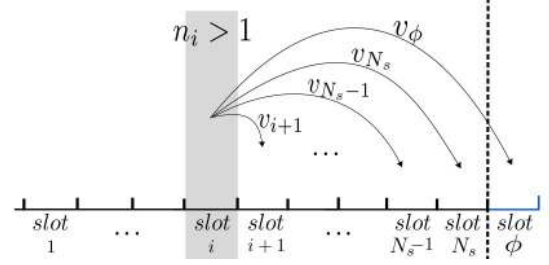

Fig. 4: Final states: n_i stations collide in slot i and contend over the remaining $N_s - i + 1$ slots. A station that chooses slot ϕ won't succeed its attempt in the current period.

The virtual slot ϕ:

According to rule *Rule 4*, if a station fails an RSS attempt, it can attempt again in the same period. However, if there are not enough slots, the RSS operation is postponed to the next period – *Rule 5*. As depicted in Fig. 4, in this case, we can imagine that, if by the end of A-BFT the station's transmission backoff counter does not reach 0, the RSS attempt will happen in a *virtual* slot ϕ outside A-BFT. Slot ϕ is useful to keep count of how many stations were not able to attempt again in the same period.

At the beginning of a period, N_a stations contend over N_s slots by uniformly selecting a number in $[0, N_s - 1]$. We define the initial state of a period as being the way stations are distributed among slots as a result of this uniform random backoff. The space of all initial possible states is $\{ n = (n_1, ..., n_{N_s}) \mid \sum_{i=1}^{N_s} n_i = N_a \}$, where n_i represents the number of stations that chose slot i. This is the set of weak compositions of N_a into N_s parts, therefore, $\binom{N_a + N_s - 1}{N_a}$ possible initial states exist [19].

Let $n = (n_1, ..., n_{N_s}, 0)$ be a possible initial state. As shown in Fig. 4, if multiple stations choose the same slot, i, a collision occurs and they are distributed uniformly among the *remaining* slots, including the virtual one. The resulting intermediate state, n', can be found using an iterative method on the number of stations, n_i, in slot i. If $n_i \in \{0, 1\}$, we simply get $n' = n$. If $n_i > 1$, let v_j for $j > i$ be the number of colliding stations that pick slot j. We get:

$$\begin{cases} n'_j = n_j & 1 \leq j < i \\ n'_i = 0 & j = i \\ n'_j = n_j + v_j & i < j \leq N_s \\ n'_\phi = v_\phi \\ \sum_{j=i+1}^{N_s} v_j + v_\phi = n_i \end{cases}$$

This operation is carried out iteratively until the end of the period. Note that the conflict is not resolved for slot N_s since colliding stations in this slot have no chance of succeeding in the ongoing period.

Let Ω be the space of all final states and, π_n, the probability of final state n. To compute the number of successes in a period, we need to construct all possible final states and count how many slots have exactly one RSS attempt. Therefore, the probability distribution of $S(N_a)$ is given by:

$$\begin{cases} \mathbb{P}(S(N_a) = k) & = 0, & k > min(N_s, N_a) \\ \mathbb{P}(S(N_a) = k) & = \sum_{n \in \Omega_k} \pi_n, & k \le min(N_s, N_a) \\ \Omega_k & = \{n \in \Omega \mid \sum_{i=1}^{N_s} \mathbb{1}_{\{n_i=1\}} = k\} \end{cases} \quad (8)$$

From (2) and (8) it is possible to derive $\mathcal{T}_{succ}(N_a)$, the success rate in a period . However, this requires the knowledge of the final state space probabilities. To this end, we introduce Algorithm 1:

Algorithm 1 Probability of final states

$\Omega \leftarrow \{n = (n_1, ..., n_{N_s}, 0) \mid \sum_{i=1}^{N_s} n_i = N_a\}$
for each state n in Ω **do**
    $\pi_n \leftarrow$ Probability of initial state n
end for
for each slot $i \in [1, N_s - 1]$ **do**
    for each state n in Ω **do**
        if slot i has $n_i > 1$ stations **then**
            $v \leftarrow$ distribute n_i over $N_s + 1 - i$ remaining slots
            $n' \leftarrow v + n$
            $\pi_v \leftarrow$ Probability of redistribution v
            $\pi_{n'} \leftarrow \pi_{n'} + \pi_n \times \pi_v$
            Remove n from Ω and add eventually n'
            $\pi_n \leftarrow 0$
        end if
    end for
end for
return $\pi_n, n \in \Omega$

(i)   It checks slot by slot for each state whether or not multiple stations chose the same slot ($n_i > 1$).
(ii) If a collision occurs in slot i, the algorithm considers all possible weak compositions v of n_i stations into $N_s - i + 1$ slots.
(iii) It computes the probability Π_v and saves the resulting intermediate state.
(iv) After all collisions are resolved (slots 1 to $N_s - 1$), all states in Ω become final states.

The state space of initial states and the subsequent intermediate states of a period become very large for high values of N_a and N_s. For this reason we carefully managed memory usage during calculations.

Using the above algorithm and Equation 8, we were able to compute π_n for all possible final states and therefore deduce the success rate in a period $\mathcal{T}_{succ}(N_a)$ with N_a active stations.

4) **Extension to non-ideal channel conditions**: The characterization of the channel quality especially in the context of mmWave communications is not trivial as many phenomena can affect the quality of communications. We present a simple model in the case where the error rate affecting the transmission of a frame is constant. Let p be the probability that a transmission will fail due to transmission errors. This error will only affect the detailed analysis at the A-BFT period scale. The only modification to our model concerns the case where a station is the only one to transmit in a slot i.e. $n_i = 1$. In this case, with a probability $1 - p$, the transmission succeeds. With

---

Algorithm 2 Probability of final states – Lossy channel

$\Omega \leftarrow \{n = (n_1, ..., n_{N_s}, 0) \mid \sum_{i=1}^{N_s} n_i = N_a\}$
for each state n in Ω **do**
    $\pi_n \leftarrow$ Probability of initial state n
end for
for each slot $i \in [1, N_s]$ **do**
    for each state n in Ω **do**
        if slot i has $n_i > 1$ stations and $i < N_s$ **then**
            $v \leftarrow$ distribute n_i over $N_s + 1 - i$ remaining slots
            $n' \leftarrow v + n$
            $\pi_v \leftarrow$ Probability of redistribution v
            $\pi_{n'} \leftarrow \pi_{n'} + \pi_n \times \pi_v$
            Remove n from Ω and add eventually n'
            $\pi_n \leftarrow 0$
        end if
        if slot i has exactly 1 station **then**
            with a probability p do $n_i \leftarrow 0$ and empty slot i
        end if
    end for
end for
return $\pi_n, n \in \Omega$

---

a probability p, we have to apply the method for analysing losses described in the previous section. Finally, the last slot, N_s, needs to be considered in the case of a lossy channel since even if only one station attempts it RSS it may fail due to channel errors. The modified approach is presented in Algorithm 2, and evaluated in Section IV-A4.

IV. Performance Evaluation

The performance evaluation is organized in two parts. In the first part, we use computer simulations to validate our analytical model. In the second part, we assess the performance of IEEE 802.11ad beamforming training using the analytical model.

A. Model Validation

Methodology: We conduct Monte Carlo simulations to validate our mathematical model. To this end, we developed a discrete-event time-driven custom simulator in Python that implements the A-BFT medium access protocol [18]. It closely follows the IEEE 802.11ad protocol details for each independently transmitting station. Unless otherwise specified, we use the default values recommended by the standard for the number of slots per period, the maximum number of successive attempts and the idle backoff window (see Table III). All simulation results are obtained with a 95% confidence interval lower than 0.004. We consider both, a loss-free physical channel where RSS failures are due to collisions only and a lossy channel where RSS failures are also due to channel errors. We model an RSS as a single frame transmission occupying one slot. We abstract the feedback mechanism (section II) and instead keep count of the number of colliding stations in each slot to determine RSS failures.

slots per period (N_s)	8
max. attempt ($MaxA$)	8
idle backoff win. ($MaxI$)	8

TABLE III: A-BFT default parameters.
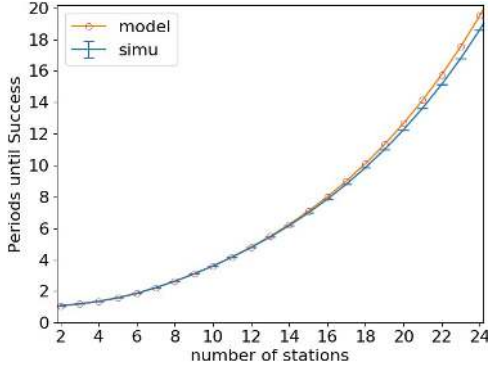
Fig. 5: Analysis vs. Simulation: The average number of A-BFT periods until an RSS succeeds, $\mathbb{E}(T_1)$.

1) **A-BFT access delay**: The objective of our model is to evaluate the A-BFT access delay: the average number of A-BFT periods – equivalent to the number of beacon intervals – until a station successfully transmits its Responder Sector Sweep (RSS). Thus, we first focus on this metric.

Fig. 5 shows the average number of periods until an RSS succeeds ($\mathbb{E}(T_1)$) , as computed by our model and via simulations. The data leads to two findings: First, which is also the main objective of this experiment, the analytical model is very accurate – analytical and simulation results practically coincide. For high loads ($16 < N < 24$), for example, the gap is less than 0.7 periods. This result indicates that the key assumption of our model – assuming the probability of successful RSS to be constant and independent of the number of A-BFT periods (Section III-C2) – has a negligible effect on its accuracy. Intuitively, this can be explained by the fact that the assumption has a bigger impact when a station takes increasingly longer to succeed its RSS, an event that is increasingly less likely.

A second finding also emerges: for dense networks, using the default parameters recommended by the standard, it may
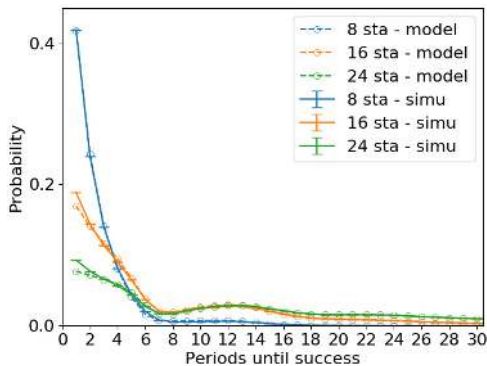

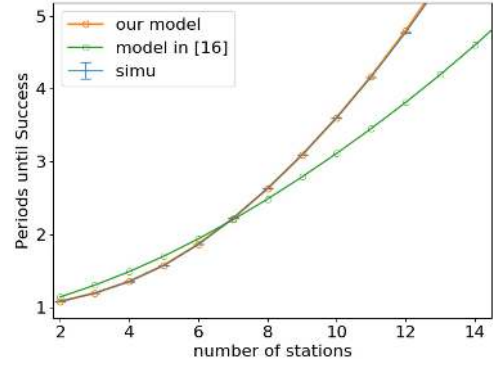Fig. 6: Analysis vs. Simulation: Access delay distribution under different loads.

Fig. 7: Access delay performance of the A-BFT protocol as calculated by our model, the model in [16] and via simulations.

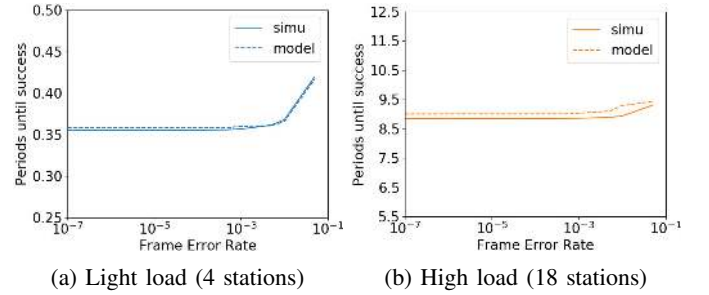(a) Light load (4 stations)   (b) High load (18 stations)

Fig. 8: A-BFT access delay with a lossy channel

take a station over a dozen beacon intervals to successfully execute an RSS – a relatively high number. We elaborate more on this in section IV-B.

2) **A-BFT access delay probability distribution**: For a more fine-grained evaluation of our model, we consider the *distribution* of the A-BFT access for different number of stations. Fig. 6 shows that the simulation and analytical model yield very similar results for any number of stations, further validating the model. What is more, an interesting behaviour emerges from the data. Under a light load ($N/N_s < 1$), the probability distribution function of the A-BFT access delay follows the trend of a geometric distribution with parameter $\frac{1}{\mathbb{E}(T_1)}$. But not under higher loads. This is due to stations becoming idle more often when the number of contending stations increases (Fig. 9d). The duration after which an idle station transitions back to active state is determined using a uniform random selection, hence the bump in the access delay distribution under high loads.

3) **Comparison with state of the art**: We compare the results of our model to the one introduced in [16], the closest work to ours to the best of our knowledge. The authors modeled a simplified version of the A-BFT protocol in which stations are allowed to attempt an RSS only once per period. As a result, Fig. 7 shows that their model fails to accurately capture the A-BFT access delay, particularly under high loads ($N/N_s >> 1$).

4) **Performance under channel errors**: We assess the impact of the channel errors on the accuracy of our model, under light and high loads. Fig. 8 shows that our model accurately captures the behaviour of A-BFT, even in the
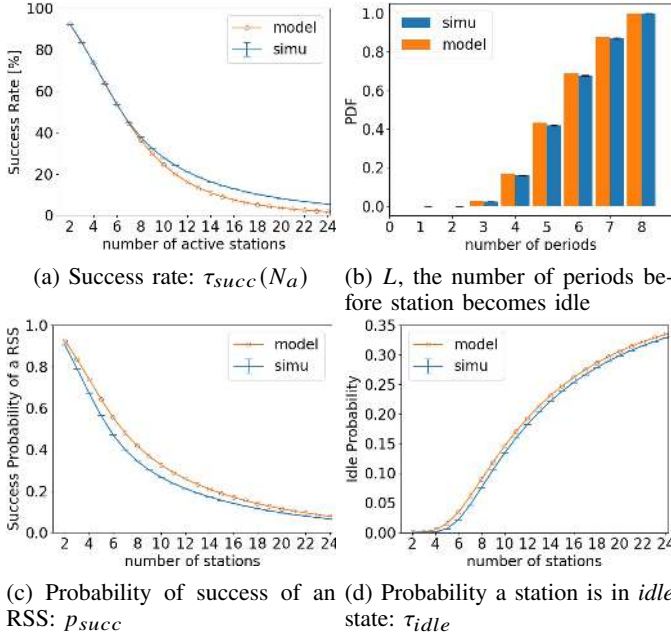
(a) Success rate: $\tau_{succ}(N_a)$

(b) L, the number of periods before station becomes idle

(c) Probability of success of an RSS: p_{succ}

(d) Probability a station is in *idle* state: τ_{idle}

Fig. 9: Analysis vs. Simulation: Components of the model using the standard's default values.

Fig. 10: The virtue of quiting easily: While the standard recommends 8 as the default value for $MaxA$, quitting after 4 failures leads to succeeding significantly earlier in dense networks.

Fig. 11: The virtue of being lazy: While the standard recommends 8 as the default value for $MaxI$, staying idle for longer leads to succeeding significantly earlier in dense networks.

presence of channel induced errors. Furthermore, the data shows that the channel errors increase the access delay. This can be explained by the fact that channel errors induce more RSS failed attempts, increasing the probability of a station becoming idle and, as a consequence, the time it takes to succeed its RSS.

5) **Model components analysis***:* Next, we focus on the different components of our analytical model: the probability distribution of L, the success rate, τ_{succ}, the success probability of an RSS, p_{succ}, and the probability of a station being in idle state, τ_{idle}.

Fig. 9 shows that the calculations of the analytical model closely match what we observe in the simulations. In the case of the probability distribution of L, Fig. 9b, and τ_{idle}, Fig 9d, the results show almost a perfect match. The difference observed in Fig. 9a is due to the pessimistic approximation we use in our model. Namely, we consider that the number of active stations stays constant within the same A-BFT period. Thus, we do not take into account the stations that reach the maximum number of consecutive failed attempts and stop contending before the period ends. As a result, the success rate given by our model is lower than that of the simulation for a high number of nodes. It is interesting to point out that especially for a high number of nodes Fig. 9c shows that the model's calculation of p_{succ} matches better the simulations. This can be explained by the fact that the model slightly overestimates τ_{idle}, Fig. 9d. As τ_{succ} and τ_{idle} both contribute to the value of p_{succ}, the model's underestimation of the former is compensated by its overestimation of the latter, leading to the observed result.

B. Assessing IEEE 802.11ad beamforming training

In this section, we use our model to analyze the performance of the IEEE 802.11ad beamforming training and more
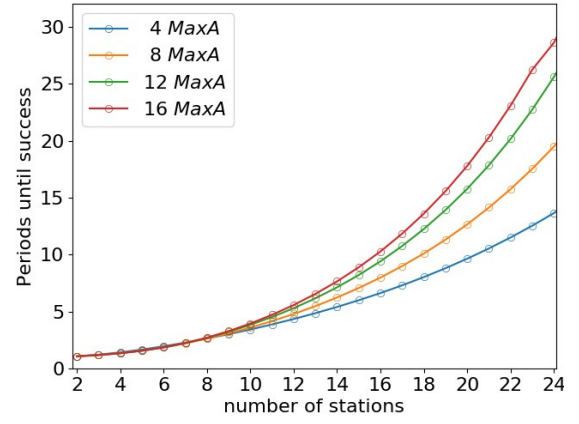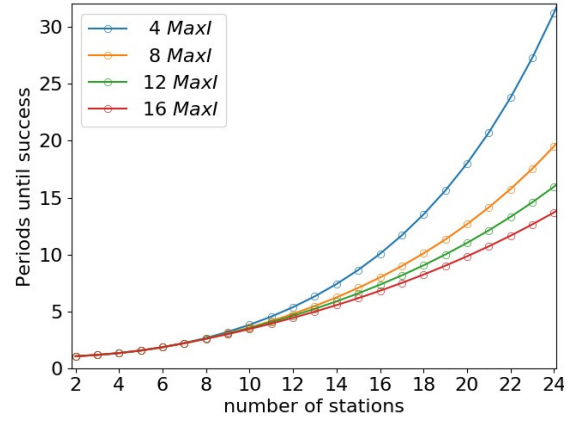
specifically the Association Beamforming Training (A-BFT), its most challenging component.

We consider the time it takes for a station to successfully execute its Responder Sector Sweep (RSS) the key performance metric of A-BFT. Therefore, we focus our attention on assessing how the three A-BFT parameters – $MaxA$, $MaxI$, N_s – affect this metric.

Methodology: To assess the impact of the three A-BFT parameters on its performance we proceed by first fixing the values of two of the parameters to those recommended by the 802.11 standard. We then use our analytical model to compute $\mathbb{E}(T_1)$ for different values of the third parameter and different numbers of stations. Recall from our Markov Chain, Fig. 3, that T_1 is the time of the first return to A_1, the state to which a station transitions after it succeeds its RSS.

Impact of $MaxA$: Fig. 10 shows the number of A-BFT periods necessary for a station to succeed its RSS as a function of $MaxA$ and the total number of stations. Two interesting findings emerge. The first is counter-intuitive – for a high number of stations, using a small $MaxA$, thus abandoning the effort to transmit the RSS more easily and transitioning to
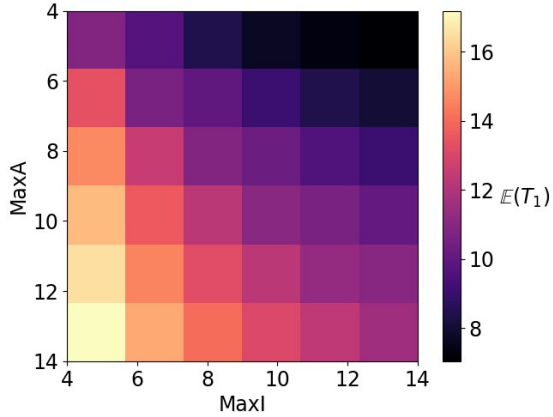
Fig. 12: Performance of A-BFT as function of MaxI and MaxA for dense networks ($N > N_s$). The Quit-easily/Be-lazy approach offers the best performance.
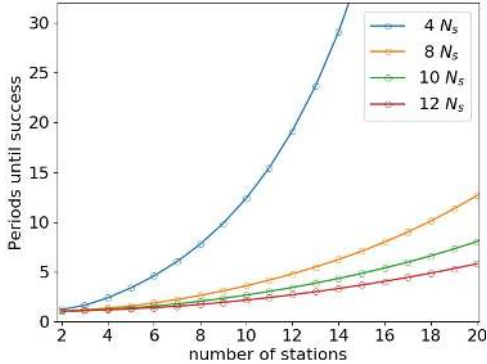

Fig. 13: Increasing the period size leads to succeeding early. However, it also reduces the time reserved for data transmission (DTI) in each beacon interval.

idle, leads to the quickest success. Second, we find that using the standard-recommended value of 8 for $MaxA$ increases by almost 40% the time it takes for a station to succeed its RSS in dense topologies.

Impact of $MaxI$: Fig. 11 shows the number of A-BFT periods necessary for a station to succeed its RSS as function of $MaxI$ and the total number of stations. The conclusions the data point to follow a similar pattern with what we observed in Fig. 10. Counter-intuitively, accepting to spend a longer time in idle mode leads a station to succeed its RSS faster in denser topologies. A $MaxI$ of 16 divides by more than two the time it takes for a station to succeed its RSS when compared with using a $MaxI$ of 4. Furthermore, the standard-recommended value of 8 for $MaxI$ is clearly not the best choice.

Quit-easily/Be-lazy: To further elucidate the findings above, in Fig. 12 we depict the performance of A-BFT as function of $MaxA$ and $MaxI$ for dense networks. The data shows that indeed combining the smallest value of $MaxA$ – Quit-easily – with the highest value of $MaxI$ – Be-lazy – leads to the best A-BFT performance. This seemingly counter-intuitive result can be explained by a careful examination of the A-BFT medium access rules (section II-B). As per Rules 1 and 4, an active station always backs off using N_s as contention window,

a constant value that does not adapt to congestion. When congestion does occur (for $N > N_s$), this mechanism does nothing to relieve it. Thus, retrying for a high number of times, a high $MaxA$, is counterproductive. In idle mode, on the other hand, stations back off using $MaxI$ as contention window, Rule 7. A high value of $MaxI$ will relieve congestion. Therefore, a policy of quiting active mode easily (small $MaxA$) and using a high backoff value while in idle (high $MaxI$) leads to the best A-BFT performance.

Impact of N_s: We assess the impact of the size of A-BFT on its performance. Fig. 13 shows that, as expected, an A-BFT with a larger number of slots enables a station to succeed its RSS faster. Note, however, that since the duration of A-BFT period is proportional to N_s, any increase to N_s will reduce the size of the DTI, the period dedicated to data transmissions. Indeed, currently the standard limits the size of N_s to 8 [2]. $MaxA$ and $MaxI$, on the other hand, only impact A-BFT, making them the most useful parameters for optimizing its performance.

V. RELATED WORK

Evaluation of IEEE 802.11ad: As the first IEEE mmWave technology available off-the-shelf, 802.11ad has been the subject of multiple performance evaluation studies. A large body of work makes use of off-the-shelf 802.11ad hardware to carefully evaluate different aspects of the protocol and/or networks constructed around it [7]–[13]. However, none of these works have looked at the performance of A-BFT (Association beamforming training protocol).

Certain works [20]–[23] have introduced analytical models to evaluate IEEE 802.11ad by extending the classic model by Bianchi [24]. However, all these studies are focused on the CSMA part of the DTI (Data Transmission Interval). A-BFT uses no carrier sensing or exponential backoff and, thus, cannot be evaluated by these models. Extending these works to model A-BFT is not straightforward.

Evaluation of IEEE 802.11ad A-BFT: A few works have focused on the evaluation of the A-BFT part of 802.11ad, however, it has been mostly using simulations [14], [15]. To the best of our knowledge, only [16], [17] introduce analytical models to study A-BFT. [16] analyzes the success probability of an RSS. However, it makes several simplifying approximations regarding the protocol: a) it assumes a station attempts at most one RSS per A-BFT period, when according to the standard [2] a station makes as many attempts as possible; b) it assumes the size of A-BFT adapts to the number of stations when according the standard it is fixed; c) it replaces the uniform backoff procedure a station performs when entering idle state with simply using its expected value. Our work makes no such approximations. Furthermore, unlike our work, their study does not consider access delay performance under various network loads. The authors in [17] study the probability of collision during A-BFT, however, it is limited to the case in which the number of stations is lower than the number of time slots in A-BFT. Our analytical model places no restriction on the number of stations.

Evaluation of other wireless access protocols: There is a rich literature on the performance of wireless access

protocols, some going back to the classic works on Aloha [25], from which IEEE 802.11ad A-BFT was inspired. However, the introduction by A-BFT of the dual active-idle race and the fact that A-BFT is interrupted regularly by the other access periods, including DTI, makes this prior research non-applicable. For example, [26] introduced a finite population model to evaluate the delay-throughput performance under stable and unstable [27]–[29] slotted Aloha channel conditions. However, the constant retry probability assumption as well as the joint assumption on active population size and transmission probability are not applicable to A-BFT due to the *idle states*. The zero-order approximation of the infinite population model in [25], [30], extended in [31], describes the maximal capacity of a slotted Aloha channel based on a Poisson arrival process. These assumptions, however, are not valid for modeling medium access in the A-BFT because the number of contending stations in the A-BFT is finite and depends on the failure history.

VI. Conclusion

We presented an analytical model for assessing the performance of beamforming training in IEEE 802.11ad networks. Our model is simple yet models accurately the details of A-BFT, the non-trivial part, protocol-wise, of the 802.11ad beamforming. Once we demonstrated the accuracy of the model using a simulator we have made available as open-source, we evaluated the performance of A-BFT as a function of its three parameters. Our results showed that a counter-intuitive, quite-easily/be-lazy approach leads to the best beamforming training performance.

Nevertheless our model presents limitations that merit additional reflection and work in the future. The two key parameters $MaxA$ and $MaxI$ are assumed by our model to take the same values for all stations while the standard allows every station to select different values. Furthermore, the model does not take into account the case in which the maximum number of SSW frames allowed to be transmitted in a slot (16 currently in the standard [2]) is not sufficient to cover all sectors.

Acknowledgment

References

[1] M. Dahhani, A. Beylot, and G. Jakllari, "An analytical model for assessing the performance of IEEE 802.11ad beamforming training," in *IFIP Networking 2020, Paris, France, June 22-26, 2020*, pp. 307–315.

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

[3] (2016) Tp-link talon ad7200. [Online]. Available: tp-link.com/us/products/details/cat-9_AD7200.html

[4] S. Aggarwal, A. Thirumurugan, and D. Koutsonikolas, "A first look at 802.11ad performance on a smartphone," in *ACM mmNets '19*, p. 13–18.

[5] S. Jog, J. Wang, J. Guan, T. Moon, H. Hassanieh, and R. R. Choudhury, "Many-to-many beam alignment in millimeter wave networks," in *NSDI '19*. USENIX Association, p. 783–800.

[6] A. Zhou, T. Wei, X. Zhang, and H. Ma, "Fastnd: Accelerating directional neighbor discovery for 60-ghz millimeter-wave wireless networks," *IEEE/ACM Transactions on Networking*, pp. 2282–2295, 2018.

[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 '15*, pp. 1–13.

[8] 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 '18*, pp. 1–9.

[9] 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 '18*, pp. 1–11.

[10] M. Dahhani, G. Jakllari, and A. Beylot, "Association and reliability in 802.11ad networks: An experimental study," in *the 44th Conference on Local Computer Networks (LCN)*, 2019, pp. 398–405.

[11] T. Wei and X. Zhang, "Pose information assisted 60 ghz networks: Towards seamless coverage and mobility support," in *ACM MobiCom '17*, p. 42–55.

[12] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Wifi-assisted 60 ghz wireless networks," in *ACM MobiCom '17*, pp. 28–41.

[13] 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 '17*, pp. 1–10.

[14] A. Akhtar and S. C. Ergen, "Efficient network level beamforming training for ieee 802.11ad wlans," in *SPECTS '15*, pp. 1–6.

[15] P. Zhou, X. Fang, Y. Fang, Y. Long, R. He, and X. Han, "Enhanced random access and beam training for millimeter wave wireless local networks with high user density," *IEEE Transactions on Wireless Communications*, pp. 7760–7773, 2017.

[16] W. Wu, Q. Shen, K. Aldubaikhy, N. Cheng, N. Zhang, and X. Shen, "Enhance the edge with beamforming: Performance analysis of beamforming-enabled wlan," in *IFIP WiOpt '18*, pp. 1–6.

[17] (2016) Channel access in a-bft over multiple channels. [Online]. Available: mentor.ieee.org/802.11/dcn/16/11-16-0101-00-00ay-channel-access-in-a-bft-over-multiple-channels.pptx

[18] M. Dahhani. (2019) IEEE 802.11ad a-bft beamforming training simulator in python. [Online]. Available: github.com/mohammeddahhani/A-BFT-beamforming-802.11ad/

[19] R. P. Stanley, *Enumerative Combinatorics: Volume 1.* Wadsworth Brooks/Cole, 1986.

[20] M. P. R. S. Kiran and P. Rajalakshmi, "Saturated throughput analysis of ieee 802.11ad edca for high data rate 5g-iot applications," *IEEE Transactions on Vehicular Technology*, pp. 4774–4785, 2019.

[21] K. Chandra, R. V. Prasad, and I. Niemegeers, "Performance analysis of ieee 802.11ad mac protocol," *IEEE Communications Letters*, pp. 1513–1516, 2017.

[22] Q. Chen, J. Tang, D. T. C. Wong, and X. Peng, "Directional cooperative mac protocol design and performance analysis for ieee 802.11ad wlans," *IEEE Transactions on Vehicular Technology*, pp. 2667–2677, 2013.

[23] C. Hemanth and T. G. Venkatesh, "Performance analysis of contention-based access periods and service periods of 802.11ad hybrid medium access control," *IET Networks*, pp. 193–203, 2014.

[24] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[25] N. Abramson, "The aloha system: Another alternative for computer communications," in *Proceedings of the November 17-19, 1970, fall joint computer conference*, ser. ACM AFIPS, p. 281–285.

[26] L. Kleinrock and S. Lam, "Packet switching in a multiaccess broadcast channel: Performance evaluation," *IEEE Transactions on Communications*, pp. 410–423, 1975.

[27] S. Lam and L. Kleinrock, "Packet switching in a multiaccess broadcast channel: Dynamic control procedures," *IEEE Transactions on Communications*, pp. 891–904, 1975.

[28] A. Carleial and M. Hellman, "Bistable behavior of aloha-type systems," *IEEE Transactions on Communications*, pp. 401–410, 1975.

[29] B. Metcalfe, "Steady-state analysis of a slotted and controlled aloha system with blocking," *ACM SIGCOMM*, p. 24–31, 1975.

[30] L. G. Roberts, "Aloha packet system with and without slots and capture," *ACM SIGCOMM Comput. Commun. Rev.*, p. 28–42, 1975.

[31] S. Lam and L. Kleinrock, "Packet switching in a multi-access broadcast channel in a computer network, Ph.D. Dissertation," *University of California at Los Angeles*, p. chapter 4, 1974.

Mohammed Dahhani obtained the M.Sc. and Ph.D. degrees in Computer Science and Telecommunications from Toulouse INP/ENSEEIHT in 2017 and 2020, respectively. His research interests are in the field of wireless networks with emphasis on mmWave protocols for 5G, access layer protocols and peformance evaluation using theoretical and experimental approaches.

André-Luc Beylot received the Ph.D. in Computer Science from the University of Paris 6 in 1993. In January 2000, he received the habilitation à diriger des recherches from the University of Versailles. Since September 2000, he is a professor in the Computer Science and Telecommunications Department of Toulouse INP/ENSEEIHT. From January 2015 to December 2020, he was the head of GDR CNRS - Rseaux et Systèmes Distribués, RSD. Since January 2021, he a is deputy director of IRIT Lab. His research interests focus on performance and design of wireless networks

Gentian Jakllari was born in Skrapar, Albania. After graduating from Petro Nini Luarasi high school in Tirana, Albania he moved to Greece where he obtained the bachelor's degree in Computer Science from the University of Ioannina in 2002. In 2003 he moved to the USA where he obtained the M.Sc. and Ph.D. degrees in Computer Science from the University of California, Riverside in 2005 and 2007, respectively. From 2007 to 2011 he was a scientist at the Internetworking Research Department at BBN Technologies in Cambridge, MA, USA. In 2011 he joined Toulouse INP-ENSEEIHT where he is now a Professor of Computer Science and Telecommunications. He holds a joint appointment with the Toulouse Institute for Computer Science Research (IRIT) where he is the head of RMESS team and co-leads the Smart City Strategic Application Area. His research interests are in computer networks and ubiquitous computing with applications to gigabit wireless networks, mobile health, IoT, smart environments, indoor localization, intelligent parking, etc.