This item is the archived peer-reviewed author-version of:

CoVRage : millimeter-wave beamforming for mobile interactive virtual reality

CoVRage: Millimeter-Wave Beamforming for Mobile Interactive Virtual Reality

Jakob Struye *Student Member, IEEE*, Filip Lemic *Member, IEEE* and Jeroen Famaey *Senior Member, IEEE*

Abstract—**Contemporary Virtual Reality (VR) setups often include an external source delivering content to a Head-Mounted Display (HMD). "Cutting the wire" in such setups and going truly wireless will require a wireless network capable of delivering enormous amounts of video data at an extremely low latency. The massive bandwidth of higher frequencies, such as the millimeter-wave (mmWave) band, can meet these requirements. Due to high attenuation and path loss in the mmWave frequencies, beamforming is essential. In wireless VR, where the antenna is integrated into the HMD, any head rotation also changes the antenna's orientation. As such, beamforming must adapt, in real-time, to the user's head rotations. An HMD's built-in sensors providing accurate orientation estimates may facilitate such rapid beamforming. In this work, we present coVRage, a receive-side beamforming solution tailored for VR HMDs. Using built-in orientation prediction present on modern HMDs, the algorithm estimates how the Angle of Arrival (AoA) at the HMD will change in the near future, and covers this AoA trajectory with a dynamically shaped oblong beam, synthesized using sub-arrays. We show that this solution can cover these trajectories with consistently high gain, even in light of temporally or spatially inaccurate orientational data.**

Index Terms—**Analog beamforming, mmWave, virtual reality, mobility, sub-arrays**

I. INTRODUCTION

MOBILE, interactive and collaborative Virtual Reality (VR), where a wireless connection provides a Head-Mounted Display (HMD) with an audiovisual stream recorded or generated in real-time, requires a reliable low-latency connection with massive throughput [1]. While some HMDs can generate content on-device, applications requiring high-fidelity, high-framerate, and high-resolution content call for a connection to a powerful machine. Current devices, such as the stand-alone Meta Quest 2, provide this capability through a wired connection or a $5\,\mathrm{GHz}$ wireless home network. The former restricts the player's freedom of movement and poses a tripping hazard, hindering immersion [1], while the latter induces noticeable video compression.

To achieve truly wireless connected HMDs, millimeter-wave (mmWave) networking is most often considered. Operating at frequencies of 30 to $300\,\mathrm{GHz}$, mmWave is capable of meeting the VR requirements [1]. Solutions often rely on the existing IEEE 802.11ad/ay Wi-Fi standards for mmWave [2], [3] or on 5G NR's mmWave capabilities [4]. The main challenges in building such a system stem from mmWave's inherently high

path loss and attenuation. To achieve sufficiently high signal strength at the HMD, the transmitter and the HMD must both focus their energy towards each other, both when sending and when receiving, in a process called beamforming. MmWave transceivers usually implement beamforming using phased antenna arrays, consisting of many separate, individually phase-controllable antenna elements [5]. By carefully tuning each element's phase shift, all elements' signals become phase-aligned, and interfere constructively, in some intended direction. 3D beamforming, with two degrees of freedom, requires a 2D grid of antenna elements, such as a Uniform Rectangular Array (URA). Beamforming is generally highly challenging, as devices are unaware of other devices' relative positions, often requiring some sort of search algorithm [6]–[8]. In a 5G context, pose estimations from the UE have been used to facilitate initial access [9] and digital beamforming [10]. For HMDs specifically, accurate on-device pose estimation [11] enables direct beamforming between the mobile HMD and static Access Point (AP). Several works have leveraged this pose information for beamforming [7], [12]–[14], noting that HMD-side beamforming is considerably more challenging than at the AP side. An angular beam misalignment of a few degrees can have a significant impact on Signal-to-Noise Ratio (SNR) [12], which is most rapidly triggered by head rotations [7]. All of the solutions only generate a beam in a single direction, based on a current pose estimate. During rapid and sudden movement, these algorithms may not be able to update their beam quickly enough. In addition, pose estimates may not be entirely accurate during such motion. As such, we pose that, to maintain mobile VR's network requirements even under rapid motion, HMD-side receive beamforming should proactively synthesize an oblong beam of dynamic width expected to continuously cover the AP during motion. In this work, we present *coVRage*, the first pose-aware receive-side beamforming algorithm to cover both the current orientation and the predicted trajectory of near-future orientations.

To enable rapid synthesis of such beams, we use sub-arrays. By dividing the full antenna array into separate sub-arrays, each sub-array can form its own sub-beam in a distinct direction. Several approaches to forming these sub-arrays exist. With physical sub-arrays, each has its own RF chain. On the other hand, virtual sub-arrays allow for dynamically configurable sub-array layouts. This class is further subdivided into analog arrays, with one RF chain for the full array, and digital arrays, with a separate RF chain for each element [15]. Virtual sub-arrays enable runtime control over the number of sub-arrays and their shape, in turn facilitating hierarchical codebooks [8], [16], 3D beams of flexible shape [17]

J. Struye and J. Famaey are with IDLab - Department of Computer Science, University of Antwerp - imec, Antwerp, Belgium (email: firstname.lastname@uantwerpen.be)

F. Lemic is with i2Cat Foundation, Spain and University of Antwerp - imec, Belgium (email: filip.lemic@i2cat.net)

(a) Localized sub-arrays   (b) Interleaved sub-arrays   (c) Multi-block sub-arrays
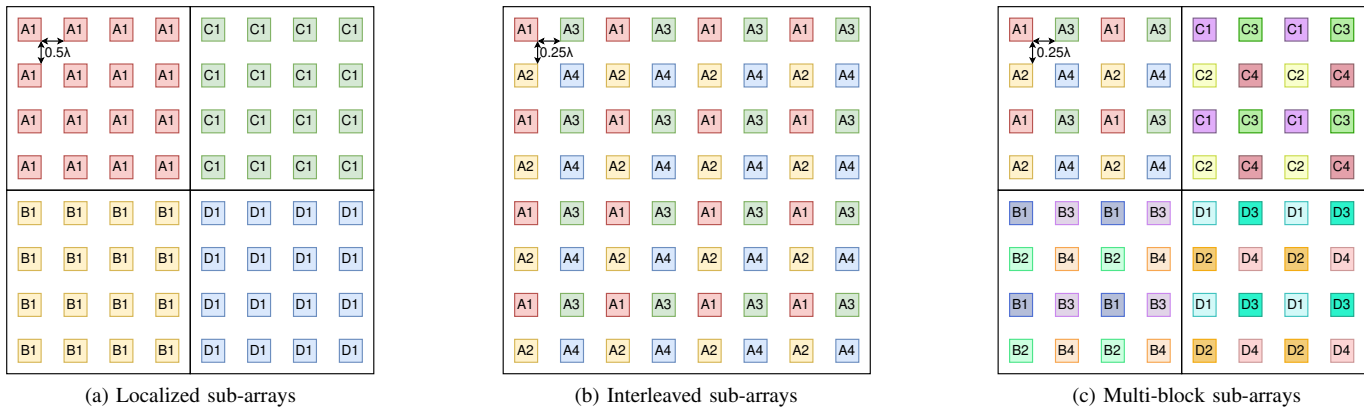
Fig. 1. Different (virtual) sub-array layouts in a phased array. A black square designates a block, which is the maximal region within which all sub-arrays are interleaved with each other. Any two sub-arrays from different blocks are localized to each other. The indicated inter-element spacings are chosen to maintain an inter-element spacing of 0.5λ within a sub-array. Each element label consists of the index of its block (a letter) and the index of its sub-array within that block (a digit).

and multiplexing/multi-user transmissions [18]. Our algorithm leverages this feature using analog sub-arrays. We avoid fully digital arrays due to their prohibitively high cost and power consumption for a mobile device.

These sub-arrays, either physical or virtual, can be distributed across the array in several ways. Specifically, a sub-array may be localized, with all elements in the sub-array next to each other, or interleaved, with elements spread across the entire array, illustrated in Fig. 1a and 1b [19]. Alternatively, a two-level *multi-block* array consists of localized blocks, each containing interleaved sub-arrays, as in Fig. 1c. Our algorithm relies on virtual multi-block arrays, which combine features of both localized and interleaved sub-arrays. Interleaved sub-arrays allow for antenna array elements placed closer together physically, utilizing limited HMD space more optimally. This reduced spacing may result in adjacent elements influencing each other's signals noticeably, such that mitigation techniques against this *mutual coupling* may be needed.

To synthesize a beam covering current and future directions towards the AP, head rotations must be predicted. Built-in pose estimation present on modern HMDs, mainly used for offsetting the expected latency of image generation, may already suffice for this application [20]. Should more accurate prediction be needed, other existing approaches can be applied. For head rotation prediction, many works have shown the effectiveness of classical approaches such as autoregression and Kalman filters [11], [21]–[23]. More recently, the field of viewport prediction, which aims to solve essentially the same problem, has successfully applied more advanced approaches such as graph learning [24], reinforcement learning [25] and recurrent neural networks [26]. While the different approaches are difficult to compare directly due to varying prediction horizons and datasets, most approaches provide predictions amply accurate for coVRage. In [11], a relatively straightforward predictor achieves a median error of under $0.35°$ at a $100\,\mathrm{ms}$ horizon for a dataset with aggressive motion, with a worst case error under $1.7°$. The dynamic width of our algorithm's beam is therefore not only intended to provide robustness against measurements errors, but also both temporal and spatial errors

in prediction.

The main contributions of this paper are summarized as follows:

1) We analyze the main challenges and opportunities specific to mmWave wireless interactive VR. We empirically estimate the expected maximal user motion, showing that rapid receive-side beamforming with orientation prediction is crucial to ensure consistently high SNR, which is facilitated by accurate pose measurements gathered by on-device sensors.

2) We present *coVRage*, a receive-side beamforming algorithm for HMDs, supporting uninterrupted connectivity during rapid head movements. This is, to the best of our knowledge, the first HMD-focused beamforming method offering proactive Angle of Arrival (AoA) trajectory coverage through sub-arrays.

3) We observe that aiming localized sub-arrays in similar directions causes them to reduce each other's beamwidths, and propose the *transitional sub-array* system to mitigate the effect. Using these, we present several mappings of sub-beams to configurable sub-arrays depending on predicted AoA changes and confidence in said predictions.

4) Using simulation, we demonstrate that coVRage and transitional sub-arrays achieve their design goals, together offering consistently high SNR during worst-case head rotations, even when these are imperfectly predicted.

The remainder of this paper is organized as follows. Section II presents the system and channel models used in this work. Section III describes the considered problem in detail. Next, Section IV presents the coVRage algorithm, and in Section V we evaluate how well it performs in simulation. Finally, Section VI concludes this paper.

Throughout the article, we use the following notation: $\mathbf{A}$ is a matrix, $\mathbf{a}$ is a vector, a is a scalar. $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ denote transpose and conjugate transpose. $\angle c$ denotes the angle of complex value c. $\mathbf{q}^*$ is the complex conjugate (and inverse) of unit quaternion $\mathbf{q}$. $|\cdot|$, $\|\cdot\|$ and $\|\cdot\|_{\mathrm{F}}$ denote the absolute value,

ℓ^2-norm and Frobenius norm. $\mathbf{I}$ is the identity matrix. $\mathbb{E}[\cdot]$ denotes the expectation.

II. SYSTEM AND CHANNEL MODEL

In this work, we consider a mmWave communication system with, at each end, a square URA consisting of N_T and N_R elements at transmit and receive side respectively. As cost and energy efficiency are of prime importance to mobile consumer devices, we assume that each array is driven by a single RF chain, restricting us to analog beamforming. For each element, the phase is individually controllable within the continuous range $[0, 2\pi)$. More realistic n-bit phase control is modeled by uniformly quantizing phase shifts [27]. Furthermore, each receive element has 1-bit amplitude control, meaning each element can be turned on or off individually [8]. As the sub-array system is purely virtual, it does not appear in the system model. We present a full system model taking reflections into account, and a simpler Line-of-Sight (LoS)-only model which is faster to simulate.

A. Full model

The full system and channel model works as follows. Given a transmitted symbol s, with unit power, the received signal becomes

$$y = \sqrt{P}\mathbf{w}_R^H \mathbf{H}\mathbf{w}_T s + \mathbf{w}_R^H \mathbf{n} \tag{1}$$

where P is the transmit power, $\mathbf{w}_R \in \mathbb{C}^{N_R}$ and $\mathbf{w}_T \in \mathbb{C}^{N_T}$ are the receive and transmit Antenna Weight Vectors (AWVs) respectively, $\mathbf{H} \in \mathbb{C}^{N_R \times N_T}$ is the channel matrix, and $\mathbf{n}$ is the Gaussian noise vector with power N_0, s.t. $\mathbb{E}[\mathbf{nn}^H] = N_0\mathbf{I}$. For an N-element square URA, the AWV is

$$\mathbf{w} = \nu \left[w_{0,0}, w_{1,0}, \ldots, w_{\sqrt{N}-1,\sqrt{N}-1} \right]^T \tag{2}$$

where ν is a normalization factor s.t. $\|\mathbf{w}\| = 1$ and

$$w_{x,y} = \alpha_{x,y} \exp(j\phi_{x,y}) \tag{3}$$

where $\alpha_{x,y} \in \{0, 1\}$ and $\phi_{x,y} \in [0, 2\pi)$ denote amplitude and phase control, respectively.

As channel model, we employ a geometric model, as is commonly used for mmWave channels [8], [28], [29]. Based on the observation that mmWave experiences limited scattering, the model considers a fixed number of paths, or Multi-Path Components (MPCs). Given the directional nature of mmWave, each MPC is identified by a physical Angle of Departure (AoD) (θ_d, ψ_d) and AoA (θ_a, ψ_a), with θ and ψ denoting azimuth and elevation, respectively. The MPC gains are Rician-distributed, with one dominant LoS MPC along with zero or more reflected Non Line-of-Sight (NLoS) MPCs [30]. The channel matrix is then written as

$$\mathbf{H} = \mu \sum_{l=0}^{L-1} \gamma_l \mathbf{a}(N_R, \theta_{a,l}, \psi_{a,l}) \mathbf{a}(N_T, \theta_{d,l}, \psi_{d,l})^H \tag{4}$$

where μ is a normalization factor $\sqrt{N_T N_R}$ s.t. $\mathbb{E}\left[\|\mathbf{H}\|_F^2\right] = N_T N_R$, L is the number of MPCs, γ_l is the complex co-efficient of the l-th path, $\mathbf{a}(\cdot)$ is the steering vector, and $(\theta_{d,l}, \psi_{d,l})$ and $(\theta_{a,l}, \psi_{a,l})$ are the AoD and AoA of the l-th

path respectively. For every NLoS path ($l > 0$), the complex coefficient γ_{nlos} is complex Gaussian distributed, and the components of the AoD and AoA are uniformly distributed in $[0, 2\pi)$. If $L > 1$, the K-factor $\mathcal{K}$ controls the LoS path's relative power, as

$$|\gamma_{los}|^2 = \mathcal{K}(L-1)\mathbb{E}[|\gamma_{nlos}|]^2 \tag{5}$$

ensuring the LoS MPC's power is expected to be $\mathcal{K}$ times as strong as all NLoS MPCs' combined. The MPCs' complex coefficients are normalized s.t.

$$\mathbb{E}\left[\sum_{l=0}^{L-1} |\gamma_l|^2\right] = 1 \tag{6}$$

The steering vector, a function of the array dimensions and the AoA or AoD, is expressed as

$$\mathbf{a}(N, \theta, \psi) = \sqrt{\frac{1}{N}} \left[a_{0,0}, a_{1,0}, \ldots, a_{\sqrt{N}-1,\sqrt{N}-1} \right]^T \tag{7}$$

where

$$a_{x,y} = \exp\left(j\frac{2\pi d}{\lambda}(x\sin\theta\cos\psi + y\sin\psi)\right) \tag{8}$$

where d is the inter-element spacing in both x and y-direction and λ is the wavelength.

In this system, the beamforming gain is expressed as

$$G = \left|\mathbf{w}_R^H \mathbf{H}\mathbf{w}_T\right|^2 \tag{9}$$

B. Simplified model

While the system and channel model presented above provides a realistic representation of the actual system, it does imply a practical inconvenience for this work. Specifically, it requires that a physical layout and AWV for both receive and transmit antenna arrays be determined, even though our work is a receive-side solution, independent of the transmit side's specifics. We therefore additionally present a simplified, transmit-independent model. For this, we first consider a LoS-only version of the full model (i.e., $L = 1$). If the transmit beam is aimed perfectly in the LoS direction, then $\mathbf{a}(N_T, \theta_{los}, \psi_{los})^H\mathbf{w}_T = 1$, with $(\theta_{los}, \psi_{los})$ being the AoD of the LoS path. In this case, the AWV and steering vector can be eliminated from (1) and (4) respectively. When re-introducing NLoS paths, the elimination of these factors alters the model's properties, as $\mathbf{a}(N_T, \theta_{d,l}, \psi_{d,l})^H\mathbf{w}_T \leq 1$ for each lth NLoS path ($l > 0$). Eliminating the factors impacts the distribution of NLoS path power in two ways: the mean increases, and the variability decreases. Within the model however, these two effects can be offset by increasing $\mathcal{K}$, and increasing the variance of the complex Gaussian distribution γ_{nlos} is sampled from, respectively. As such, we argue that the following simplified model still represents the expected environment with sufficient accuracy, while being both faster to compute, and independent of the transmit antenna array's dimensions. The following two equations comprise the simplified model:

$$y = \sqrt{P}\mathbf{w}_R^H \mathbf{H}s + \mathbf{w}_R^H \mathbf{n} \tag{10}$$

and

$$\mathbf{H} = \mu \sum_{l=0}^{L-1} \gamma_l \mathbf{a}(N_R, \theta_{a,l}, \psi_{a,l}) \tag{11}$$

where μ is now $\sqrt{N_R}$. The effect of a suboptimally aimed transmit beam can still be modeled by increasing N_0 and decreasing $\mathcal{K}$. With this model, the beamforming gain becomes

$$G = \left| \mathbf{w}_R^H \mathbf{H} \right|^2 \tag{12}$$

When only considering the LoS path (i.e., choosing $\mathcal{K} \to \infty$, meaning that $\mathbb{E}\left[\, |\gamma_{nlos}| \, \right]^2 \to 0$ and $|\gamma_{los}|^2 \to 1$), we can write this as

$$G = \left| \mu \mathbf{w}_R^H \mathbf{a}(N_R, \theta_{los}, \psi_{los}) \right|^2 \tag{13}$$

in which case the beamforming gain equals the receive gain. We consider this single-beam model as sufficiently realistic, as LoS is unlikely to be broken for the mobile VR use case this work considers. Furthermore, reflections in said use case are most likely via walls, and their power is assumed to be negligible as long as the user is not right next to the wall and grating lobes are avoided. Redirected walking can keep mobile users away from walls in VR applications [31].

III. BEAMFORMING FOR MOBILE VIRTUAL REALITY

In this section, we outline the challenges and opportunities of mmWave VR, describe the environment we envision our algorithm to operate in, and present the phased array layouts used for this work.

A. Challenges and Opportunities

In this work, we consider a VR setup where a ceiling-mounted mmWave AP serves a mobile VR user on the ground. The user moves with Six Degrees of Freedom (6DoF), and the HMD generates high-accuracy estimates of its own pose (i.e., location plus orientation) and can generate short-term pose predictions with reasonable accuracy [20], [32]. To ensure a high Quality of Experience (QoE), the downlink from AP to HMD must have a consistently high SNR, even during rapid user movement. From a beamforming perspective, this application is unique on several fronts. In this section, we outline the three main fronts, and discuss the challenges and opportunities they imply for mmWave HMD beamforming.

First are the mobility patterns of HMDs. For any APs involved, the location and orientation are static and known a priori. HMD-wearing users, however, are often encouraged to move around rapidly, especially in truly wireless VR. Gauging the impact of both translational (e.g., walking) and rotational (e.g., head rotations) motion using realistic velocities obtained from an experiment shows that rotations are significantly more impactful. Consider an AP $2\,\mathrm{m}$ above an HMD moving translationally at a walking pace of $1.4\,\mathrm{m/s}$. In $100\,\mathrm{ms}$, this will shift the optimal beam direction by $4°$ for both the AP and HMD. Instead, a rotational motion of $300\,°/\mathrm{s}$ may, depending on the direction, cause a shift in optimal beam direction of up to $30°$ in $100\,\mathrm{ms}$ for the HMD, with minimal impact on the AP. In Sec. V-A, we observe that users can reach a maximum velocity over three times as high in a $100\,\mathrm{ms}$ window. Clearly,

HMD-side beamforming under rotational motion is the most challenging sub-problem of mobile VR beamforming.

A second front encompasses VR network requirements in terms of latency, throughput and reliability. Depending on other sources of latency in the full content delivery chain, the target network latency may be between 1 and $5\,\mathrm{ms}$ [1]. This implies that, when delivering a live video stream per-image, the interval between subsequent images may be larger than the maximum latency, meaning only a fraction of this interval should be used for transmission, inflating the throughput requirements. Specifically, a $1\,\mathrm{Gbps}$, $100\,\mathrm{Hz}$ video stream requires a $10\,\mathrm{Gbps}$ connection to be delivered at a $1\,\mathrm{ms}$ per-image latency [33]. Non-uniform image sizes due to encoding further inflate this requirement. Finally, per-image reliability must also be high. Packet loss can cause image degradation or even loss of an entire image, severely reducing the QoE. This traffic pattern is known as High Rate, High Reliability and Low Latency Communications (HR2LLC) [34]. To provide such performance, consistently high-gain links are essential.

A final front involves the HMD's sensors. A modern HMD can sense its own position and orientation accurately in real-time. Modern HMDs, such as the Meta Quest 2, can generate these measurements without additional equipment, through *inside-out* tracking. This information can be leveraged for fast, accurate beamforming. General beamforming solutions commonly rely on some sort of search algorithm to determine a high-gain beam, possibly from a predefined set of beams [6]–[8]. As the steering vector is easily derived from pose information, a beam focused in the LoS direction can be synthesized directly and efficiently by simply setting the AWV in (2) to the steering vector in (7). This maximizes the gain in (13), as $\left|\mathbf{b}^H \mathbf{a}\right| \leq 1$ and $\mathbf{a}^H \mathbf{a} = 1$ for any normalized $\mathbf{a}$ and $\mathbf{b}$.

These specifics of interactive wireless VR over mmWave indicate the challenges and opportunities for designing an application-specific beamforming solution. As receive-side beamforming, especially during fast rotations, is significantly more challenging than transmit-side beamforming, this work focuses on this first type. While first-generation IEEE 802.11ad hardware often foregoes receive-side beamforming, instead employing quasi-omnidirectional receive beams [35], this is inefficient from a link SNR perspective [36], as receive-side beamforming reduces interference [37]. This is especially important in multi-AP scenarios, where different APs may serve different nearby users. As such, transmit-only beamforming is not expected to achieve the SNR needed to fulfill the HR2LLC requirements. Furthermore, theoretical quasi-omni beams are unrealistic, with gain across their supposedly flat range often varying significantly [38], [39].

B. Antenna Array Design

The antenna array for the HMD should be designed with the expected environment in mind. This subsection provides general design guidelines.

First of all, we eliminate hybrid and digital arrays. To enable *non-analog* beamforming on a battery-powered device, lower-resolution analog-to-digital and digital-to-analog converters are necessary [40]. This reduces achievable SNR, making

it undesirable for this application. While non-analog arrays are a requirement to send different data streams in different directions, sending/receiving the *same* data stream in/from different directions is perfectly possible with an analog array, using virtual sub-arrays. The potential benefit of sending multiple data streams to the same HMD using non-analog arrays is known to be limited [41]. As such, we opt to rely on analog HMD-side arrays.

A next trade-off to consider is between the number of elements in the array, and the spacing between these elements. For a square N-element URA, the attainable beamwidth in radians, for both azimuth and elevation, is approximately

$$b_\alpha = \frac{0.886\lambda}{\sqrt{N}d\cos\alpha} \qquad (14)$$

with α the steering angle for either azimuth (θ) or elevation (ψ), $\alpha = 0$ being broadside, and d the inter-element spacing [42]. A square URA is evident, as the x and y-directions of the array, with broadside towards the ceiling, are essentially interchangeable for a user able to rotate around their own axis. The beamwidth equation implies that, for a fixed physical area, adding more elements within said area will not tighten the beamwidth. As such, an inter-element spacing of $d = 0.5\lambda$ is often used, as a tighter spacing leads to impractically large beams, while wider spacing is known to create *grating lobes*; undesired side lobes with a directional gain as high as the main lobe's [42]. This rule of thumb, however, no longer applies when using interleaved sub-arrays. If only each ath element belongs to the same sub-array, the inter-element spacing within a sub-array effectively becomes ad, as illustrated by Fig. 1b. As such, the physical inter-element spacing should be chosen with a specific a in mind. Furthermore, for sufficiently large arrays, the multi-block layout, as previously illustrated in Fig. 1c, becomes feasible.

IV. CoVRage

In this section, we provide a step-by-step explanation of how coVRage works. We first cover how to determine the directions of the sub-beams, how to synthesize them and how to avoid destructive interference along the predicted trajectory between them. Next, we present how to map the sub-beams to the available sub-arrays, which is of crucial importance for multi-block arrays. Finally, we outline the computational complexity of the approach.

A. Sub-beam Generation

CoVRage must convert measured current and predicted future HMD orientations to a set of phase shifts for the phased array in the HMD. We decompose this process into three distinct steps. First, we determine how the AP appears to move relative to the HMD, the reference point. Specifically, we determine the direction of the AP at the start and end of the rotation between HMD orientations, and the shortest trajectory between these directions. Next, we determine a set of beams that covers this trajectory, achievable by the phased array. Finally, we minimize the destructive interference between the sub-arrays on the trajectory, to avoid having "blind" spots

along the trajectory. In this section, we represent rotations (and orientations) as *unit quaternions* (e.g., $\mathbf{q}$), as they are easily composed and interpolated.

1) Trajectory Generation: In the physical environment, the AP is a static object, while the HMD and its antenna array rotate. From the antenna array's point of view however, the AP then appears to rotate around it. In other words, we consider the horizontal coordinate system (i.e., elevation-azimuth system) with its origin in the $(0,0)^{\text{th}}$ array element, as was used for beamforming in Section II. Every real-world head movement then rotates this coordinate system. In the field of 3D kinematics, this is called a *passive* (or *alias*) transformation [43]. For coVRage, an *active* (or *alibi*) transformation, in which we assume the AP moves around the HMD, is more convenient. Negating the rotational angle converts between active and passive transformations. In other words, when the HMD rotates from position $\mathbf{q}$ to $\mathbf{p}$, the coordinate system performs a passive rotation $\mathbf{pq}^*$, equivalent to the AP (and any other static objects in the room) performing the active rotation $(\mathbf{pq}^*)^* = \mathbf{qp}^*$. To translate rotations to the AP's absolute direction, the AP direction at one instant must be known. This can be pre-configured, or measured using existing AP sensing approaches [14].

As the HMD is only expected to measure and predict the start and end of the expected rotation within some brief time-frame, coVRage is responsible for generating the path of the AP direction during the rotation, between those two points. Given a human head's moment of inertia [44], we will assume that the most likely trajectory between two orientations is the one with the shortest angular distance. This single rotation, known to exist from Euler's rotation theorem, is easily calculated when the end points are represented as quaternions, using the Slerp algorithm [45]. When rotating from $\mathbf{q}$ to $\mathbf{p}$, the set of quaternions $(\mathbf{pq}^*)^a$ for $a \in [0,1]$ covers exactly all orientations along the shortest trajectory between the start and end orientations.

2) Sub-Beamforming: Once the AP trajectory as seen from the HMD is determined, the algorithm needs to synthesize a beam covering it. For this step, we represent directions not as the commonly used azimuth-elevation pair (θ, ψ), but rather in UV-coordinates (u, v) with $u = \cos\psi\sin\theta$ and $v = \sin\psi$ [7]. While beamwidth depends on the angular distance from broadside for azimuth-elevation (see (14)), the beamwidth is nearly invariant to the beam's direction with UV-coordinates [46]. Specifically, the beamwidth in UV-space can be approximated by the constant

$$b_{uv} = \frac{0.886\lambda}{Nd} \qquad (15)$$

with an error always under 2%, highest near the edges of the hemisphere. As shown in Fig. 2, a rectangular sub-array's beam anywhere in UV-space is as such accurately represented by a circle of constant radius, eliminating the need for complicated, time-consuming beam shape calculations. As conversion from quaternions, used in the previous step, to UV-coordinates, is not a linear transformation, equidistantly sampled quaternion orientations are not necessarily equidistant in UV-space, and are not necessarily on a straight line. Still, the
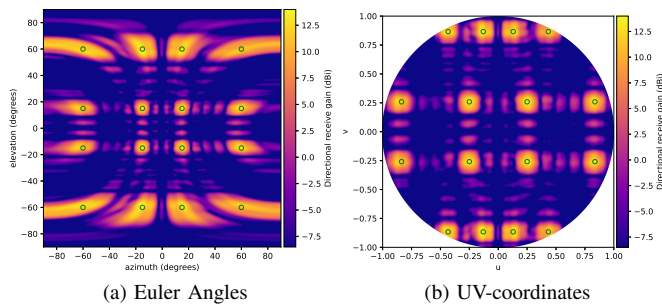
(a) Euler Angles

(b) UV-coordinates

Fig. 2. 16 beams, with θ and $\psi \in \{-60, -15, 15, 60\}$ degrees, all appear as near-perfect circles in UV-space, but are warped significantly near the edges in Euler coordinates. Green circles show the intended directions.

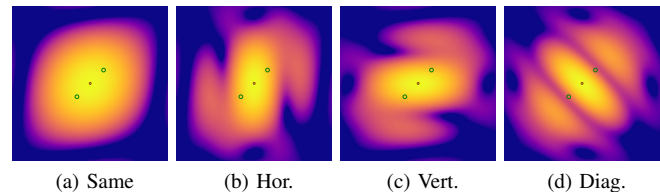(a) Same   (b) Hor.   (c) Vert.   (d) Diag.

Fig. 3. Aperture fusion is clearly visible when forming the same two adjacent sub-beams (large green circle) with the same midpoint (small brown circle) with sub-arrays either in the same block, or in horizontally/vertically/diagonally different blocks.

problem of trajectory coverage with sub-beams is essentially reduced to covering a curve using circles. We can sample points along this curve, but the total length of the curve is not (accurately) known a priori. As such, we propose the following sampling approach. We begin by sampling orientations with $a := 0$, increasing a by some δ_a for every next sample. In addition, we set some static target UV-space distance between samples $\bar{\delta}_s$. For every new sample, we update

$$\delta_a := \delta_a \frac{\bar{\delta}_s}{\delta_s} \qquad (16)$$

with δ_s the actual distance between the two most recent samples. For a sufficiently small $\bar{\delta}_s$, this approach generates a near-equidistant list of directional samples along the trajectory. Using these samples, we now easily estimate the total trajectory length t and the desired angular distance between two adjacent sub-beams Δ_s as

$$\Delta_s = \frac{t}{s-1} \qquad (17)$$

where s is the number of sub-beams available. By using $s-1$ rather than s, the centers of the first and last sub-beam, rather than their edges, can coincide with the start and end of the trajectory. Experimentation showed that this was necessary to achieve sufficient gain near the start and end of the trajectory. It is then straightforward to aim the sub-beams by iterating over all sampled trajectory points and aiming a sub-beam towards one whenever the desired distance Δ_s is reached. The points halfway between each adjacent pair of sub-beams are called *midpoints* and are stored for the following step. Note that an array can only cover a trajectory if $\Delta_s \leq b_{uv}$.

The optimal AWV $\mathbf{w}$ pointing to (θ, ψ), ignoring NLoS paths, is easily derived from the channel model. The gain in some AoA (θ, ψ) is maximized by setting the AWV (before normalization) equal to the steering vector for that AoA, defined in (7). For the full AWV of coVRage's entire beam, which aims at several directions, it suffices to, for each element (x, y), set its AWV element $w_{x,y}$ to the steering vector for the AoA of the sub-beam formed by the sub-array that element belongs to. In other words, the beamforming parameters of $w_{x,y}$ shown in (3) are set to $\alpha_{x,y} := 1$ and

$$\phi_{x,y} := \frac{2\pi d}{\lambda} \left(x \sin \theta_s \cos \psi_s + y \sin \psi_s \right) \qquad (18)$$

when the element at position (x, y) contributes to the sub-beam aimed at (θ_s, ψ_s).

3) Sub-Beam Syncing: Once the sub-array layout is determined and each sub-array is aimed properly, the remaining step is to synchronize the sub-beams, eliminating destructive interference between sub-arrays along the trajectory. As global optimization at this level is challenging and expensive, we apply a heuristic inspired by previous work on analog sub-arrays [17]. Specifically, we minimize destructive interference between adjacent sub-beams where it is expected to be the most impactful. In this case, this is the point along the trajectory equidistant from the two sub-beam directions. These midpoints were already stored during the previous step. To maximize constructive interference between adjacent sub-beams at the midpoint, coVRage calculates the phase difference between the two, and applies this as an additional phase shift to all elements contributing to the second sub-beam, making the two sub-beams phase-aligned at the midpoint. The phase of a sub-beam i for midpoint $m = (\theta_m, \psi_m)$ is

$$\phi_{i,m} = \angle \left(\mathbf{w}_i^H \mathbf{a}(N_R, \theta_m, \psi_m)_i \right) \qquad (19)$$

where the subscript i indicates that all entries not corresponding to an array element contributing to sub-array i are set to 0. For all sub-arrays $i > 0$, all entries of $\mathbf{w}$ corresponding to an element contributing to that sub-array are multiplied by a phase offset

$$o_i = \exp \left(j \left(\phi_{i-1,i-1} - \phi_{i,i-1} \right) \right) \qquad (20)$$

where o_{i+1} must only be calculated after o_i was applied.

B. Sub-beam Assignment

With a fully interleaved array, the mapping of sub-beams to the available sub-arrays does not influence performance. With a multi-block array however, this becomes an important aspect of the algorithm. In a multi-block array, two sub-arrays from the same block interact as if they are in an interleaved configuration, while sub-arrays from different blocks interact as if localized. Fig. 1c shows a multi-block array configured to consist of 4 blocks of 4×4 elements each.

As (14) implies, beamwidth is inversely proportional to the array aperture, i.e., the physical size of the array. When sub-arrays behave as interleaved, combining two by aiming them in the same direction increases the total aperture by only one inter-element spacing d, regardless of array size. In the localized case however, the total aperture (roughly) doubles,
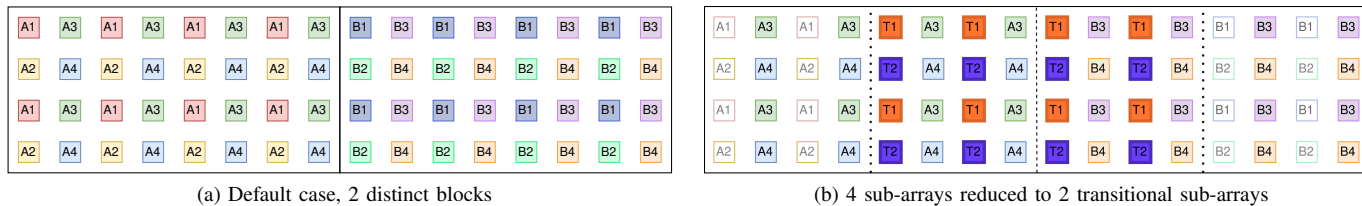
(a) Default case, 2 distinct blocks

(b) 4 sub-arrays reduced to 2 transitional sub-arrays

Fig. 4. On the left, a 16×4 antenna array is configured as two 8×4 blocks of four 4×2 interleaved sub-arrays each. To reduce aperture fusion between sub-arrays from different blocks, we introduce transitional sub-arrays in the right configuration, shown with thicker edges. Half of the elements initially assigned to sub-arrays A1 and B1 now form a new transitional sub-array T1, in a new transitional block (delimited by dotted lines), overlapping with the two original blocks (separated by dashed line). The remaining elements of A1 and B1 are disabled, shown in white. Analogously, T2 is formed from A2 and B2.

effectively halving the beamwidth. When two sub-beams are only slightly offset, this *aperture fusion* still occurs to a large extent, as Fig. 3 illustrates. CoVRage for multi-block arrays must therefore take care to avoid aperture fusion where it could cause "holes" in the coverage, but could also leverage this effect for tighter coverage if desired. We therefore propose two different approaches to multi-block beamforming: a *loose* beam where aperture fusion is avoided as much as possible as to maintain beamwidth offered by a single-block array, and a *tight* beam where aperture fusion is triggered as much as possible as to achieve higher gain exactly on-trajectory. In both cases, further measures must be taken to ensure a stable coverage of consistent width throughout the trajectory. In what follows, we assume that the considered trajectory is predominantly *horizontal*, meaning its range on the u-axis exceeds that on the v-axis. For the vertical version, simply swap the x and y-coordinates of the array elements.

We first present a method of mitigating unwanted aperture fusion. For this, we introduce the *transitional sub-array*, illustrated in Fig. 4. To form a horizontal transitional sub-array, take two sub-arrays from horizontally adjacent blocks, and combine them into a new sub-array of the same size by disabling the leftmost half of the left sub-array, and the rightmost half of the right sub-array. Disabling individual antenna elements requires 1-bit amplitude control, which is often assumed to be supported [8], [47]. Using a transitional sub-array, aperture fusion is smoothed out over a larger angular distance. This is easily derived from the beamwidth equation in (15). Consider two sub-arrays, forming two adjacent sub-beams. These will be phase-aligned at the midpoint, meaning (15) can be used to calculate beamwidth as if the involved antenna elements formed one larger sub-array aimed towards this midpoint. As such, having two sub-arrays in different blocks, instead of the same block, will reduce midpoint beamwidth by 50%. When instead using a transitional sub-array, this reduction is limited to 33%.

As sub-arrays within the same block are interchangeable, the main challenge concerns *block transitions*. Each time two adjacent sub-beams are mapped to sub-arrays in different blocks, either horizontally or vertically, a block transition occurs, triggering aperture fusion. To optimize performance, we must decide whether to minimize or maximize the number of transitions for the two directions. If aperture fusion is desirable for one direction, a transition can occur for every pair of adjacent sub-beams, and only a single transition is needed

TABLE I
LOOSE (HORIZONTAL) BEAM CONFIGURATIONS

| | Trans. sub-arrays | | |
	Hor.	Vert.	Sub-beams
Non-diag.	0	2	14
Semi-diag.	1	2	12
Diag.	1	1	13

in the undesirable direction, for a 2×2 block array. If both directions are to be minimized, a minimum of three transitions must occur, including at least one in each direction. For any undesirable transition, we must also determine an appropriate number of transitional sub-arrays. As each transitional sub-array deactivates elements and decreases the total sub-array count, this number should be minimized.

For a (horizontal) tight beam, vertical transitions tighten the beam vertically and are therefore desirable. Horizontal transitions may cause coverage holes and are therefore minimized. Extensive experimentation showed that two transitional sub-arrays were needed to consistently eliminate said holes. This leaves a total of 14 sub-beams. Fig. 5a illustrates this layout.

With a loose beam solution, all block transitions are undesirable. With horizontal trajectories, we opt for two horizontal transitions and one vertical transition, as the latter has a larger impact on coverage consistency. Careful experimentation revealed that the number of required transitional sub-arrays depends on the slope of the trajectory. We broadly identified three different cases depending on (mean) UV-space slope m of the trajectory: a *non-diagonal* trajectory with $|m| < 1/3$, a *diagonal* trajectory with $|m| > 2/3$ and a *semi-diagonal* trajectory in-between. Table I shows the number of transitional sub-arrays needed per transition, and the resulting number of sub-beams, and Fig. 5b to 5d illustrate them. As the number of transitional sub-arrays is highest in the semi-diagonal case, the other two can be seen as optimizations.

C. Computational Complexity

Given that analog beamforming with quantized phase shifters is generally NP-hard [48], we opted for a heuristic approach for coVRage, intended to run in real-time. In this subsection, we analyze the complexity of the algorithm's main steps. The complexity of trajectory generation depends on the trajectory length t and the parameter $\bar{\delta}_s$, controlling the target

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3229134

8

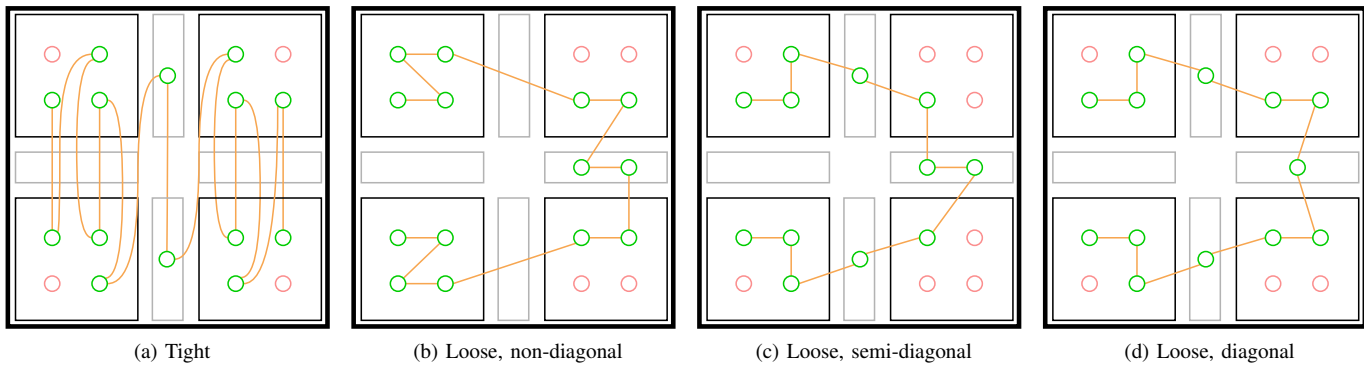| (a) Tight | (b) Loose, non-diagonal | (c) Loose, semi-diagonal | (d) Loose, diagonal |

Fig. 5. The order in which sub-arrays are mapped to sub-beams, for each type of beam, for an array such as seen in Fig. 1c. Each circle is a sub-array, consisting of several antenna elements. Each sub-array may be active (green) or inactive (pink). The orange line connects all active sub-arrays in order, such that connected sub-arrays form adjacent sub-beams. For each transitional sub-array (in a grey box), a sub-array is deactivated in each adjacent block. Within a block, sub-arrays are interchangeable.

distance between sampled points, resulting in a complexity of $O(\frac{t}{\overline{\delta}_s})$. If needed, one can increase the parameter $\overline{\delta}_s$ to reduce runtime. The only downside of this is an increase of $\frac{\overline{\delta}_s}{2}$, the worst-case angular difference between the selected and the optimal sub-beam direction. Next, sub-beam synthesis is $O(N)$ with N the number of elements, which is optimal when not relying on pre-computed beams from a codebook. Finally, sub-beam syncing is $O(s)$ with s the number of sub-beams. Based on the runtime of the current, unoptimized implementation, we fully expect the algorithm to synthesize a beam in under $1\,\mathrm{ms}$ on-HMD. As the HMD's processor should have some idle time between processing arriving content frames (once every $8.33\,\mathrm{ms}$ at $120\,\mathrm{Hz}$), we do not expect coVRage's computations to have any impact on QoE. Furthermore, as we target a prediction horizon in the order of $100\,\mathrm{ms}$, the runtime latency is not expected to impact coVRage's effectivity.

V. EVALUATION

In this section, we simulate coVRage to evaluate how well it performs in the envisioned scenario. First, we visualize its performance in individual trajectories to illustrate that it achieves its trajectory-covering goal with both loose and tight beams. Then, we evaluate its performance over many thousands of randomly generated trajectories using carefully crafted metrics to show the algorithm's consistency. In the evaluation, we consider antenna arrays that would fit into an HMD. Measuring several commercially available devices showed that a $4\,\mathrm{cm} \times 4\,\mathrm{cm}$ array is a fair assumption. For the first set of experiments, we simulate a $120\,\mathrm{GHz}$ array, resulting in a 64×64 quarter-wavelength-spaced array, divided into four blocks of four interleaved half-wavelength-spaced sub-arrays each, thereby capable of producing 16 distinct sub-beams. For these experiments, we use the simplified LoS-only model as its complexity vastly increases the number of simulations we can perform, and assume non-quantized phase shifters. Afterwards, we consider a 32×32 element array with a single block of four interleaved sub-arrays, targeting $60\,\mathrm{GHz}$, and evaluate the impact of introducing NLoS links and quantized phase shifters. All simulations are implemented in C++.

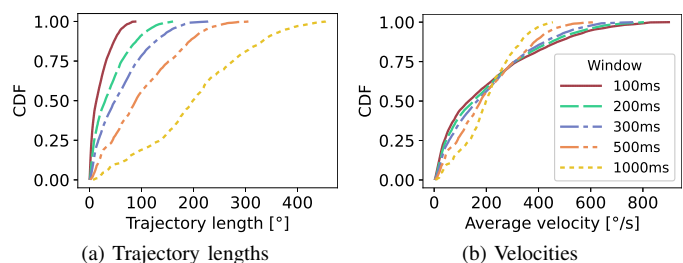
| (a) Trajectory lengths | (b) Velocities |

Fig. 6. Cumulative Distribution Function (CDF) of the angular trajectory lengths and velocities within several sliding windows, gathered from real-world measurements.

A. Individual trajectories

Before evaluating coVRage, we investigate the range of trajectories we can reasonably expect to encounter. One existing dataset on high-motion head movement shows a maximal velocity of $167\,°/\mathrm{s}$ [11]. It is however unclear if this shows instantaneous velocity, or velocity averaged over some temporal window. Another work shows speeds up to $360\,°/\mathrm{s}$ over a $100\,\mathrm{ms}$ window for only the yaw axis [7]. In addition to this existing data, we gathered a dataset focused on determining maximal realistic angular velocities. A subject wearing the Meta Quest 2 was encouraged to move quickly and erratically as much as they were comfortable with, and rotational data was logged at the maximal frequency of $972\,\mathrm{Hz}$. To mitigate any sample-to-sample inaccuracies in the measurements, we analyze the performed rotations with sliding windows of different lengths, between $100\,\mathrm{ms}$ and $1\,\mathrm{s}$. Fig. 6 shows the range of trajectory lengths and velocities produced within sliding windows of different lengths, across $45\,\mathrm{s}$ of rapid movement. This shows a maximal rotation of $160°$ for $200\,\mathrm{ms}$. Intuitively, it makes sense that a human is not able to turn their head more than $180°$ in a single rapid motion. The maximum velocity, averaged over the window, also decreases with longer windows, as the human body intuitively cannot sustain maximum angular velocity for long. At $200\,\mathrm{ms}$, the velocity peaks at $804\,°/\mathrm{s}$. We use the $200\,\mathrm{ms}$ window as a reference, as, for larger sliding windows, fast rotation was usually not sustained within the full window. Based on this,
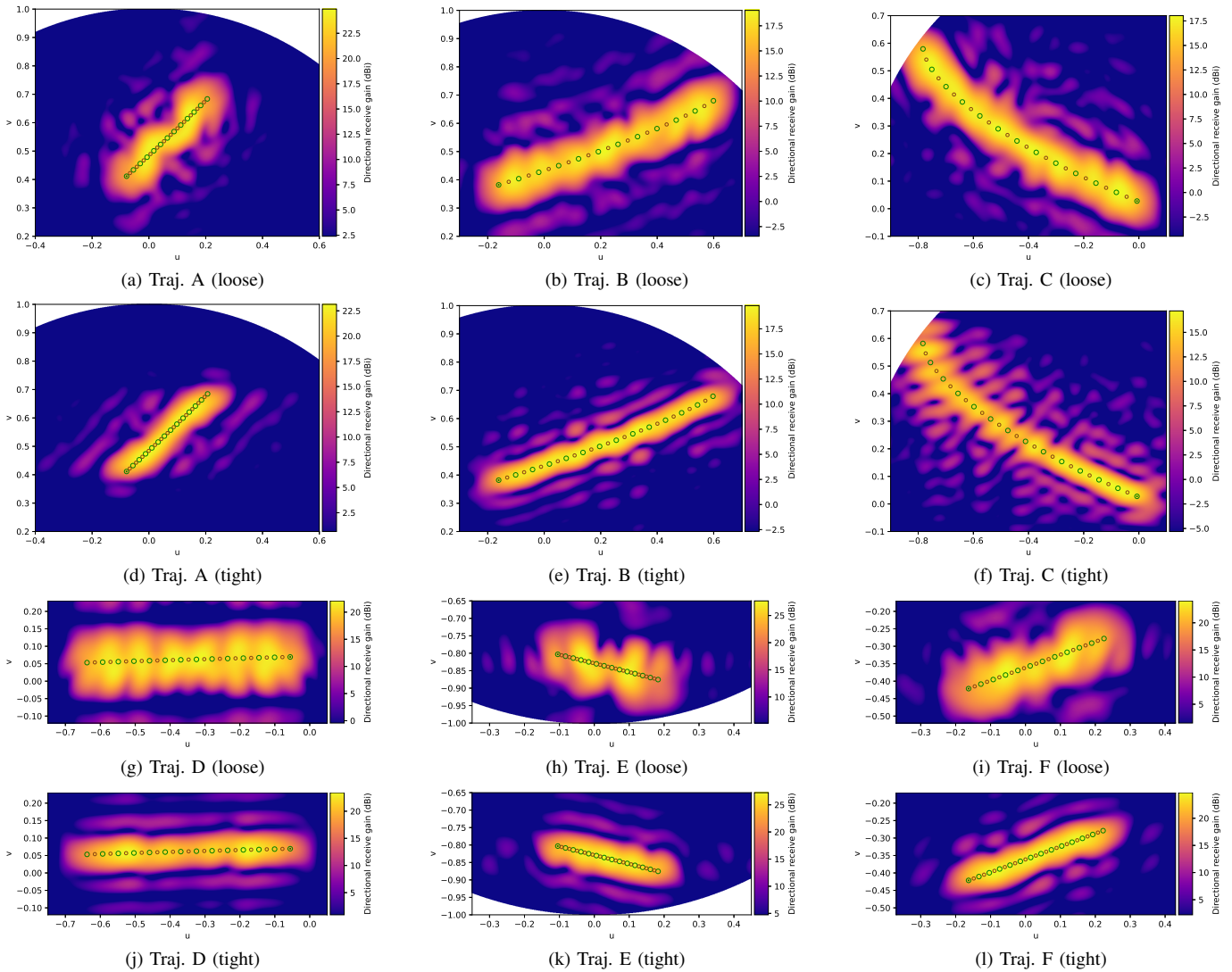
Fig. 7. Loose and tight beam coverage for a broad range of trajectories, showcasing coVRage's effectiveness, as well as known shortcomings (in 7f and 7h). Large green circles indicate sub-beam directions, with the double circle being the start of the trajectory. Small brown circles indicate the midpoints, where two adjacent sub-beams are phase-aligned. Plots are zoomed-in versions of the full-range UV plot (see Fig. 2b), with plots in the same row sharing the same zoom level.

we only consider trajectories of at most $180°$.

To visually present the performance of coVRage, we introduce the *gain map*. For some selected AWV, the gain map shows, for every possible direction, what the beamforming gain would be if this direction were the AoA. The map covers the hemisphere centered around the receive array's broadside, which points directly at the ceiling when the subject is standing up straight. Ideally, this gain should be consistently high along the trajectory and low elsewhere. For an initial analysis, we generate a broad set of gain maps, and pick a subset that shows both the general effectiveness of the algorithm, as well as its main limitations. Fig. 7 shows these gain maps for both loose and tight beams, showing that both beam types generally achieve their goals. Two trajectories were selected to highlight known limitations of the algorithm. The tight beam in Fig. 7f has reduced gain in the top half. Longer trajectories can curve noticeably in UV space, meaning one may transition from predominantly horizontal to predominantly vertical. As

sub-array order varies significantly between the two (Fig. 5a shows the horizontal version, and can be mirrored diagonally to obtain the vertical one), and only one order can be used throughout the trajectory, the tight beam does not perform well along part of the trajectory. While the tight beam was designed with either horizontal or vertical tightening in mind, it does not degrade significantly for shorter diagonal trajectories such as Fig. 7d, although the tightening effect is reduced. As loose beams take the trajectory's slope m into account, the difference in sub-array order for beams with $|m|$ slightly under 1 (Fig. 5d) and slightly over 1 (Fig. 5d mirrored diagonally) is minor: one horizontal transitional sub-array is swapped for a vertical one. As such, loose beams do not struggle in this case, as Fig. 7c shows.

Secondly, Fig. 7h and, to a lesser extent, Fig. 7a, show an inconsistent beamwidth for the loose beam. This occurs in short trajectories, because non-adjacent sub-beams may be close enough to influence each other significantly. In this case,
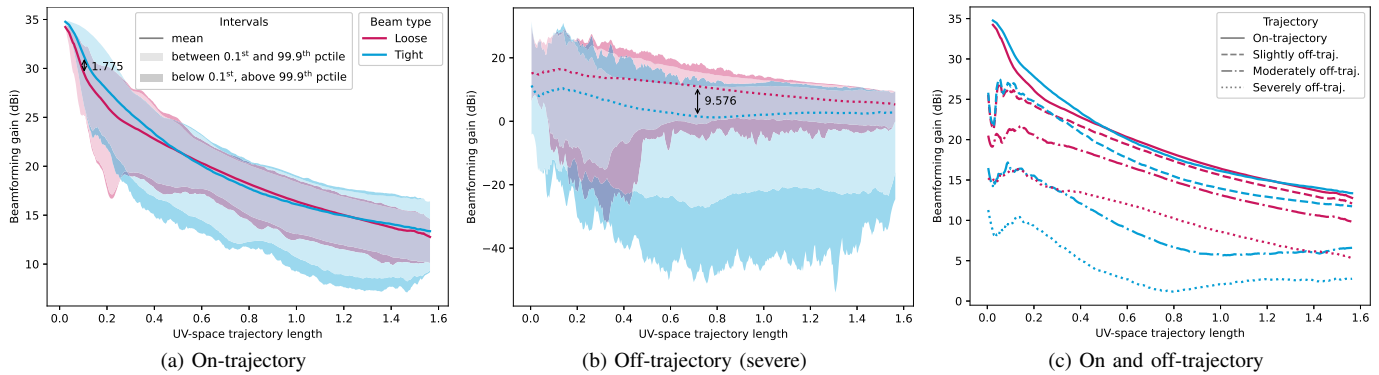
Fig. 8. Fan charts showing the mean beamforming gain along the full actual trajectory for all beam types depending on trajectory length, aggregated over $100\,000$ simulations, grouped by trajectory length. For off-trajectory, the trajectory expected while beamforming differs from the actual trajectory along which gain was measured.

the number of transitional sub-arrays introduced to mitigate aperture fusion may not suffice for very close sub-beams. Overall, the issue with tight beams only occurs for longer trajectories, while the issue for loose beams only occurs with shorter trajectories, with the two susceptible ranges being non-overlapping. As accurate measurements and predictions are more challenging under fast motion, during which erratic changes in velocity or heading are more likely to occur, loose beams are generally preferable for longer trajectories. Conversely, tight beams are generally superior for shorter trajectories, which are easier to measure and predict. As such, each beam type's issue only occurs when that beam type is not the most appropriate beam type, making these issues only a minor limitation of the coVRage approach, which can be solved by intelligently selecting the most appropriate beam type, based on parameters such as expected trajectory length and confidence in predictions.

B. Detailed analysis

Next, we evaluate whether coVRage can repeatedly achieve its consistent trajectory coverage for a wide range of trajectories. Even occasional drops in coverage are expected to have a significant negative impact on QoE. As such, we randomly generate $100\,000$ trajectories, in the $20°$ to $180°$ range. We sample the start and end points of the trajectories uniformly in the 3D rotation group $SO(3)$ using a well-known algorithm [49], [50]. We introduce three metrics which together characterize coVRage's consistency: *trajectory gain*, *gain concentration* and *gain variation*. Trajectory gain measures the gain experienced on all points along the trajectory. As implied by Parseval's theorem [51], increasing the trajectory length decreases achievable (stable) gain along said trajectory. As such, we report trajectory gain as a function of trajectory length. We subdivide the metric into on-trajectory gain, where the trajectory was perfectly measured and predicted, and off-trajectory gain, where we determine the gain along a trajectory not coinciding with the trajectory coVRage intended to cover, evaluating the impact of imperfect measurements and/or predictions. For the off-trajectory case, every point in the actual trajectory is shifted along the normal in said point

by a fixed UV-space distance. Secondly, the gain concentration is based on the linear gain integrated over a range of directions in the hemisphere. Specifically, for some angular distance δ, the gain concentration indicates the fraction of the total integrated gain that is at most δ removed from the nearest point on the trajectory. This provides an indication of beamwidth. Ideally, this function would increase near-linearly within the intended width of the beam, reaching 1 at the edge of the intended coverage zone. Any gain outside of the intended coverage could be considered "lost", following again from Parseval's theorem. Finally, gain variation is the difference between the lowest and highest gain along the trajectory. The lower the variation, the more consistent the coverage is. As coVRage operates in UV-space, the results are expressed in UV-coordinates. Small UV-coordinates are nearly equal to the azimuth and elevation in radians.

1) Trajectory gain: First, we analyze the on-trajectory gain. For this, trajectories of similar UV-length (bins of width $0.004\,\mathrm{uv}$[1]) are grouped together and all sampled gains (at granularity $0.004\,\mathrm{uv}$) are combined per-bin. For visual clarity, the data in the following graphs was smoothed using a Savitzky-Golay filter ($window = 11, order = 2$) [52], without affecting any findings.

Fig. 8a shows the results for the two beam types, shown using fan charts. In these, lines show median values, while shaded areas cover all samples, with the areas containing the lowest and highest $0.1\,\%$ shaded more darkly. The graph shows that, for trajectories with a length up to $0.5\,\mathrm{uv}$, the tight beam achieves the higher mean gain, while its 0.1^{st} percentile is better until $0.3\,\mathrm{uv}$. With longer trajectories, the tight beam begins to degrade, as discussed in Section V-A. This result indicates that the trajectory length for which the optimal beam switches from tight to loose is somewhere between $0.3\,\mathrm{uv}$ and $0.5\,\mathrm{uv}$. For short trajectories, the loose beam performs somewhat erratically, as also covered in Section V-A. In this case, a tight beam is advisable. The largest difference in gain between tight and loose beam is $1.775\,\mathrm{dB}$. This means that

---

[1]While UV-coordinates are dimensionless, we introduce the unit uv to more clearly indicate (angular) distances in UV-space, with $\mathrm{uv} \equiv 1$, analogous to the rad unit for (dimensionless) radians.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3229134
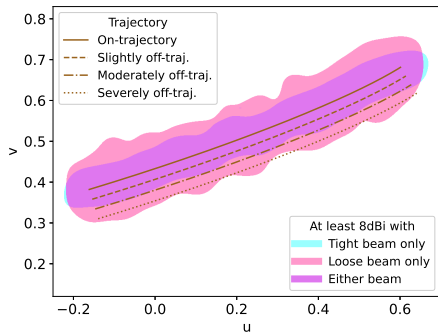
11

Fig. 9. Simplified gain map showing high coverage zones (at least $8\,\mathrm{dBi}$), overlaid with (non-)shifted trajectories
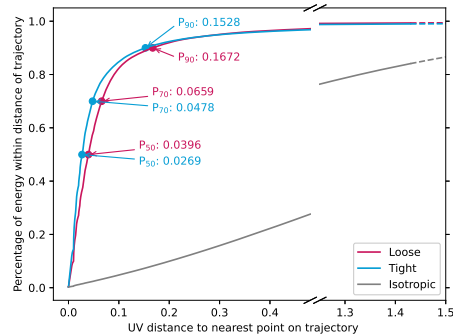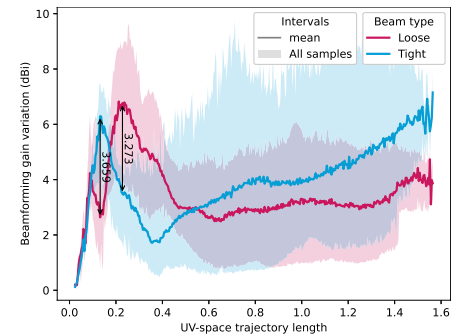
Fig. 10. Gain concentration

Fig. 11. Gain variation

the tight beam can offer a gain exceeding that of the loose beam by at most $50\,\%$, in linear terms.

Fig. 8b evaluates the off-trajectory case, with a shift of $0.075\,\mathrm{uv}$. In such a situation, the loose trajectory performs significantly better, as it achieves its design goal of covering a wider area around the expected trajectory, to account for errors in measurement and prediction. The difference in mean gain is significant for any trajectory length, reaching a maximum of over $9.5\,\mathrm{dBi}$. The tight beam reaches a 0.1^{st} percentile of under $-25\,\mathrm{dBi}$. The loose beam again performs somewhat poorly in the lowest percentiles with shorter trajectories. Starting from $0.5\,\mathrm{uv}$ the 0.1^{st} percentile does not fall below $-2.5\,\mathrm{dBi}$. As significant measurement and prediction errors are mainly expected for longer trajectories, this is only a limited issue.

Finally, Fig. 8c shows the mean gains from the previous two figures, along with less drastically shifted trajectories of $0.025\,\mathrm{uv}$ and $0.050\,\mathrm{uv}$, showing that, as intended, the loose beam already outperforms the tight beam even for minor trajectory shifts. Fig. 9 illustrates these shifted trajectories in a simplified gain map for the same beams shown in Fig. 7b and 7e. Clearly, only the loose beam covers the more significant shifts.

2) Gain concentration: Fig. 10 shows the gain concentration, averaged over all trajectories, for both loose and tight beams. For comparison, the plot also includes the gain concentration of a theoretical isotropic receive antenna. As an isotropic antenna has a constant gain for all directions, the curve equals the Cumulative Distribution Function (CDF) of a randomly chosen direction's distance to the trajectory. As the CDFs for both loose and tight beams grow significantly more rapidly than this baseline CDF, the algorithm clearly achieves its main goal of focusing energy around the trajectory.

It additionally shows that the tight beam accomplishes its goal of focusing energy more tightly around the trajectory. The largest percentage point difference is achieved in the band containing all directions at most $0.036\,\mathrm{uv}$ removed from the trajectory, which contains, on average, $60.5\,\%$ of all energy with a tight beam but only $46.5\,\%$ in the loose case. Conversely, the patch containing half of all energy is $47.2\,\%$ wider for the loose beam compared to the tight beam, as shown with percentile values in the plot. Starting from $94.4\,\%$ of all energy

however, the loose beam does overtake the tight beam. We hypothesize that this occurs due to sub-beams from interleaved sub-arrays partially fusing together, effectively reducing the inter-element spacing, which is known to reduce side lobe strength [42]. As the tight beam does perform better near the trajectory, this phenomenon does not detract from the usefulness of the tight beam. Note that nonzero energy faraway from the trajectory is entirely unavoidable with phase-shifting arrays, as phases cannot perfectly cancel each other out for a large, continuous range of directions.

For both beam types, the gain concentration for very short and long trajectories was significantly lower compared to the average case, due to coVRage's reduced effectiveness in such cases, as described above. As the relative reduction in concentration was similar between loose and tight beams, and as these cases are relatively rare, further evaluation of gain concentration as a function of trajectory length would provide no significant additional insight.

3) Gain variation: Fig. 11 shows the gain variation for the two beam types, as a function of the trajectory length. Both types experience a large peak with short trajectories, with the same cause as previous similar phenomena. Starting from $0.5\,\mathrm{uv}$, the variation for loose beams stabilizes around $3\,\mathrm{dBi}$. This indicates that coVRage achieves a rather even distribution of energy across the trajectory, with the best-covered direction along the trajectory receiving roughly twice as much energy as the worst-covered direction. Ideally, the variation would be zero, however this is again not perfectly achievable with phase shifting. The variation starts to deteriorate slightly for very long trajectories, increasing by around $1\,\mathrm{dBi}$. For tight beams, it steadily deteriorates starting from length 0.4, becoming higher than the loose beam's around $0.55\,\mathrm{uv}$, again showing that tight beams are mainly suitable for shorter trajectories.

Overall, these three metrics together show how coVRage is able to consistently distribute nearly all available energy evenly along the expected trajectory.

C. Single-block arrays

The previous results, such as Fig. 11, show that there is a range of short trajectories for which the signal strength is inconsistent. As the proximity of sub-beams is responsible for this, reducing the number of sub-beams may alleviate this
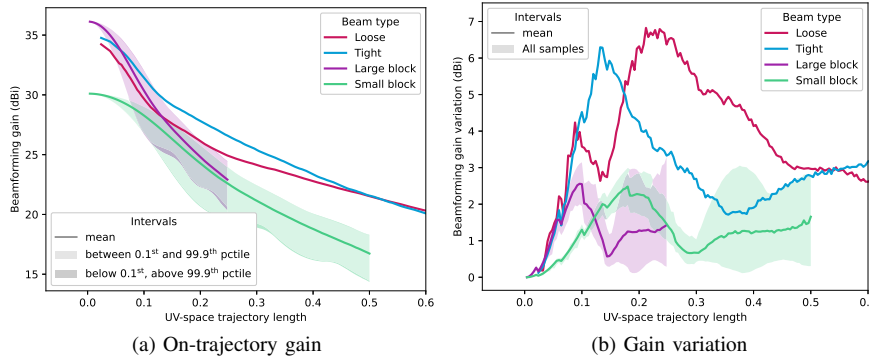
(a) On-trajectory gain   (b) Gain variation

Fig. 12. Gain metrics for single-block solutions, with one small (32×32) or large (64×64) array of four interleaved sub-arrays. The shown UV-range is reduced, and shading for multi-block solutions are removed, to focus on single-block results. Full multi-block results are shown in Fig. 8a and 11.

Fig. 13. CDF of the change in on-trajectory gain when including 2 NLoS paths in the model, averaged over 250 random initializations of said paths.

TABLE II
GAIN DIFF., QUANTIZED VS. CONTINUOUS (dB)

| | Quantization bits | | | |
	2	4	6	8
P_{99}	2.68	0.480	0.115	0.0287
$P_{99.9}$	4.95	0.758	0.181	0.0450
Worst case	59.9	1.65	0.496	0.0995

phenomenon. We consider two methods of achieving this: merging all four blocks into one *large* 64×64 block, or simply disabling three of the blocks, leaving one *small* 32×32 block. Both leave four sub-beams, with the former's being twice as tight. For each of the two solutions, we again perform $100\,000$ simulations, restricting trajectory length to between $0°$ and $30°$. In Fig. 12, we compare the on-trajectory gain and gain variation of these new results to those previously discussed. Given the lower maximal UV-space trajectory length with only four sub-beams, we restrict the results to $0.25\,\mathrm{uv}$ and $0.5\,\mathrm{uv}$ for the large and small block, respectively. Except for very short trajectories, under $0.13\,\mathrm{uv}$, both multi-block solutions outperform the large block in terms of average on-trajectory gain, while the small block never reaches the average performance of either multi-block solution.

In terms of gain variation however, both single-block solutions significantly outperform the multi-block solutions for the shorter paths considered. The two display a similar curve, although the small blocks curve is more stretched horizontally, due to its wider sub-beams. These results also show that the overall gain variation pattern, consisting of an initial peak for short trajectories, followed by a trough and a gradual increase, is inherent to the sub-beam system and cannot be avoided for a single combination of sub-beam width and number. This indicates that there may be merit to an extension of coVRage with dynamic sub-beam width and count, although we leave further investigation of this approach for future work.

D. Phase Shift Quantization

In the preceding evaluations, we assume that the phase shifters of the antenna array can continuously achieve any angle with perfect 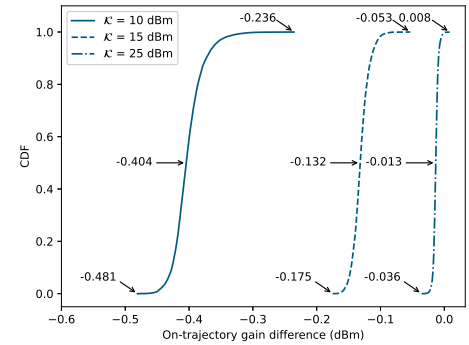accuracy. In practice, phase shifters only provide a fixed, limited set of angles. An n-bit phase shifter supports 2^n angles, leading to a granularity of $\frac{2\pi}{2^n}$. To investigate how well coVRage performs with quantized phase shifts, we perform the simulation for $10\,000$ randomly generated trajectories, with 2, 4, 6 and 8-bit phase shifts, along with a continuous baseline. For every possible direction, we compare the beamforming gain of the quantized and continuous systems. To focus on (desired or undesired) high-gain directions only, we filter out any directions where neither system achieves at least $10\,\mathrm{dBi}$, which leaves around five million data points per quantization level.

Table II shows the 99th, 99.9^{th} and worst-case absolute difference between the quantized and continuous systems. 2-bit phase shifters occasionally experience significant gain differences, making them ill-suited for achieving reliably high gain with coVRage. Starting from 4-bit, the difference is at most $1.65\,\mathrm{dBi}$, with $99\,\%$ under $0.5\,\mathrm{dBi}$. The latter is achieved for all samples with 6 bits, and an 8-bit solution further reduces the worst-case difference to under $0.1\,\mathrm{dBi}$. Considering the gain fluctuation throughout the trajectory (Fig. 11), coVRage is viable with 4-bit phase shifters, and near-indistinguishable from the continuous case with 8-bit phase shifters.

E. Multipath Model

All preceding experiments use the simplified LoS-only model due to runtime concerns. Here, we compare the results of the LoS-only and multipath models for a modest set of trajectories to show that the two produce similar results. The simplified model with NLoS paths is equivalent to the full model with a single-element isotropic transmitter. Introducing a beamformed transmitter aimed reasonably well towards the receiver would only reduce the impact of NLoS paths.

The multipath model features two additional parameters: the number of paths L and the Rician K-factor $\mathcal{K}$, representing the relative strength of the LoS path. We set L to 3, in line with related works works, e.g., [8], as the mmWave channel is known to be sparse. As mmWave K-factor values vary significantly across the literature [8], [53]–[56], we perform separate simulations with $\mathcal{K} = 10$, 15 and 25 dB. For 128 uniformly sampled trajectories, we run 250 simulations with randomly sampled path coefficients and AoAs, both with a

TABLE III
ON-TRAJECTORY GAIN DIFFERENCE WITH NLoS LINKS (dB)

	$\mathcal{K}$-**factor** (dB)		
	10	15	25
$P_{99.99}$	-5.423	-2.662	-0.7532
$P_{99.9999}$	-11.38	-5.931	-1.225
Worst case	-17.68	-10.57	-1.576

loose and a tight beam, for all three K-factors, for a total of $192\,000$ simulations. Fig. 13 shows the CDF of the gain difference for all points on all considered trajectories. It shows that, on average, the worst gain reduction experienced for any point on any trajectory was under $0.5\,$dBi for the strongest NLoS paths and under $0.04\,$dBi for the weakest. Compared to the gain fluctuation along a trajectory in Fig. 11, the impact of NLoS paths on coVRage's performance is negligible.

In the above, we average each result across 250 simulations. In contrast, Table III shows the (near-)worst-case percentiles without averaging, emphasizing outlier results. This shows that, while a significant gain reduction occur very occasionally with lower K-factors, the reduction remains modest for higher K-factors even in the worst case. As the K-factor is known to increase with increasing frequency [57], we expect the higher K-factors to give more accurate results. The negligible impact of the NLoS paths and higher runtime of the full model supports our LoS-only approach to the full evaluation of coVRage.

F. Practical Limitations

Beamforming is based on wave interference, i.e. the interaction of waves emitted by the different array elements. Nearby antenna elements will however also interact from an electromagnetic perspective, as they absorb, re-radiate, reflect and scatter each other's signals [58], [59]. This process, called mutual coupling, becomes more prominent as inter-element spacing shrinks [60] and has numerous effects, with a non-negligible (but not necessarily negative) effect on beam gain. As this effect is highly computationally complex to model and dependent on physical characteristics of the specific antenna array, the effect is customarily not considered in information-theoretic evaluations, such as ours. Numerous hardware mitigation techniques against mutual coupling, orthogonal to the beamforming process, have been proposed [58], [59].

VI. CONCLUSIONS

In this paper, we presented coVRage, the first beamforming algorithm designed specifically for HMD-side beamforming in mobile VR applications, where uninterrupted reception even during fast head rotations is crucial for maintaining QoE. Using the HMD's built-in orientation detection capabilities, a predictor can estimate how the AoA of incoming wireless video data will change in the near future (e.g., next $200\,$ms). By dynamically subdividing the phased array into virtual sub-arrays and aiming each sub-array's beam at a different point along the predicted trajectory, coVRage offers uninterrupted coverage along the full trajectory. At $120\,$GHz, coVRage

can also dynamically change the beamwidth depending on confidence in the measurements and predictions. Through simulation, we evaluated the performance of coVRage for a large range of realistic rotational trajectories. We showed that the algorithm consistently provides a high and stable gain along the trajectory. The dynamic coverage width provides a tunable robustness to temporal and spatial errors in measurements and predictions. Furthermore, we showed that coVRage remains functional with quantized phase shifters and under the influence of reflections. Overall, coVRage provides an important step towards reliable connectivity for truly wireless and highly mobile interactive VR with future HMDs supporting extremely high resolutions and refresh rates. In future work, we will extend coVRage to the scenario where multiple APs cover the HMDs [61]. In addition, we envision an enhancement using Intelligent Reflective Surfaces [62], which could provide LoS communication even when such a link is not directly available between AP and HMD, expected to occur mainly when a user looks up or down.

REFERENCES

[1] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Netw.*, vol. 32, no. 2, pp. 78–84, 2018.

[2] W. Na, N.-N. Dao, J. Kim, E.-S. Ryu, and S. Cho, "Simulation and measurement: Feasibility study of tactile internet applications for mmwave virtual reality," *ETRI J.*, vol. 42, no. 2, pp. 163–174, 2020.

[3] R. Zhong, M. Wang, Z. Chen, L. Liu, Y. Liu, J. Zhang, L. Zhang, and T. Moscibroda, "On building a programmable wireless high-quality virtual reality system using commodity hardware," in *8th Asia-Pacific Workshop Syst.*, 2017.

[4] Y. Liu, J. Liu, A. Argyriou, and S. Ci, "Mec-assisted panoramic vr video streaming over millimeter wave mobile networks," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1302–1316, 2019.

[5] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. USA: Cambridge University Press, 2005.

[6] 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 *13th Int. Conf. Emerg. Netwo. Experiments Technol.* ACM, 2017, p. 414–425.

[7] Y. Yang, A. Zhou, L. Wu, S. Xu, H. Ma, T. Wei, and X. Zhang, "Scalable 3d beam-steering for directional millimeter wave wireless networks," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2021.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3229134

14

[8] Z. Xiao, T. He, P. Xia, and X. Xia, "Hierarchical codebook design for beamforming training in millimeter-wave communication," *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3380–3392, 2016.

[9] S. Rezaie, E. de Carvalho, and C. N. Manchón, "A deep learning approach to location-and orientation-aided 3d beam selection for mmwave communications," *arXiv preprint arXiv:2110.06859*, 2021.

[10] S. Zhou, L. Chen, and W. Wang, "Attitude information aided digital beamforming in millimeter-wave mimo systems," *IEEE Syst. J.*, pp. 1–12, 2021.

[11] H. Himberg, Y. Motai, and A. Bradley, "A multiple model approach to track head orientation with delta quaternions," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 90–101, 2013.

[12] O. Abari, D. Bharadia, A. Duffield, and D. Katabi, "Enabling high-quality untethered virtual reality," in *14th USENIX Symp. Networked Syst. Des. Implementation (NSDI 17)*, Boston, MA, Mar. 2017, pp. 531–544.

[13] A. Zhou, L. Wu, S. Xu, H. Ma, T. Wei, and X. Zhang, "Following the shadow: Agile 3-d beam-steering for 60 ghz wireless networks," in *IEEE Conf. Comput. Commun. (INFOCOM)*, 2018, pp. 2375–2383.

[14] T. Wei and X. Zhang, "Pose information assisted 60 ghz networks: Towards seamless coverage and mobility support," in *23rd Ann. Int. Conf. Mobile Comput. Netw.*, 2017, p. 42–55.

[15] T. Shimura, T. Ohshima, H. Ashida, S. Ishikawa, S. Fujio, A. Honda, Z. Li, K. Nishikawa, C. Kojima, K. Ozaki, M. Shimizu, and Y. Ohashi, "Millimeter-wave tx phased array with phase adjusting function between transmitters for hybrid beamforming with interleaved subarrays," in *2016 46th Eur. Microw. Conf. (EuMC)*, 2016, pp. 1572–1575.

[16] V. Raghavan, J. Cezanne, S. Subramanian, A. Sampath, and O. Koymen, "Beamforming tradeoffs for initial ue discovery in millimeter-wave mimo systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 543–559, 2016.

[17] L. Zhu, J. Zhang, Z. Xiao, X. Cao, D. O. Wu, and X. Xia, "3-d beamforming for flexible coverage in millimeter-wave uav communications," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 837–840, 2019.

[18] H. Li, M. Li, Q. Liu, and A. L. Swindlehurst, "Dynamic hybrid beamforming with low-resolution pss for wideband mmwave mimo-ofdm systems," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 9, pp. 2168–2181, 2020.

[19] J. A. Zhang, X. Huang, V. Dyadyuk, and Y. J. Guo, "Massive hybrid antenna array for millimeter-wave cellular communications," *IEEE Wireless Commun.*, vol. 22, no. 1, pp. 79–87, 2015.

[20] S. M. LaValle, A. Yershova, M. Katsev, and M. Antonov, "Head tracking for the oculus rift," in *IEEE Int. Conf. Robot. Automat. (ICRA)*, 2014, pp. 187–194.

[21] J. J. LaViola, "A comparison of unscented and extended kalman filtering for estimating quaternion motion," in *2003 Amer. Control Conf.*, vol. 3, 2003, pp. 2435–2440 vol.3.

[22] E. Kraft, "A quaternion-based unscented kalman filter for orientation tracking," in *6th Int. Conf. Inf. Fusion*, vol. 1, 2003, pp. 47–54.

[23] A. van Rhijn, R. van Liere, and J. D. Mulder, "An analysis of orientation prediction and filtering methods for vr/ar," in *IEEE Proc. Virtual Reality*, 2005, pp. 67–74.

[24] X. Zhang, G. Cheung, Y. Zhao, P. Le Callet, C. Lin, and J. Z. G. Tan, "Graph learning based head movement prediction for interactive 360 video streaming," *IEEE Trans. Image Process.*, vol. 30, pp. 4622–4636, 2021.

[25] X. Feng, Z. Bao, and S. Wei, "Liveobj: Object semantics-based viewport prediction for live mobile virtual reality streaming," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 5, pp. 2736–2745, 2021.

[26] L. Yang, M. Xu, Y. Guo, X. Deng, F. Gao, and Z. Guan, "Hierarchical bayesian lstm for head trajectory prediction on omnidirectional images," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021.

[27] C. You, B. Zheng, and R. Zhang, "Channel estimation and passive beamforming for intelligent reflecting surface: Discrete phase shift and progressive refinement," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2604–2620, 2020.

[28] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, 2014.

[29] G. Kwon, N. Kim, and H. Park, "Millimeter wave sdma with limited feedback: Rf-only beamforming can outperform hybrid beamforming," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1534–1548, 2019.

[30] M. Jasim, M. Ababneh, N. Siasi, and N. Ghani, "Hybrid beamforming for link recovery in millimeter wave communications," in *IEEE 19th Wireless Microw. Technol. Conf. (WAMICON)*, 2018, pp. 1–4.

[31] E. R. Bachmann, E. Hodgson, C. Hoffbauer, and J. Messinger, "Multi-user redirected walking and resetting using artificial potential fields," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 5, pp. 2022–2031, 2019.

[32] T. A. Jost, B. Nelson, and J. Rylander, "Quantitative analysis of the oculus rift s in controlled movement," *Disability Rehabil.: Assistive Technol.*, vol. 0, no. 0, pp. 1–5, 2019, pMID: 31726896.

[33] J. Struye, F. Lemic, and J. Famaey, "Towards ultra-low-latency mmwave wi-fi for multi-user interactive virtual reality," *2020 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, 2020.

[34] C. Chaccour, M. N. Soorki, W. Saad, M. Bennis, and P. Popovski, "Can terahertz provide high-rate reliable low-latency communications for wireless vr?" *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9712–9729, 2022.

[35] 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 *15th Ann. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, 2018, pp. 1–9.

[36] Y. Wang, T. Zhang, S. Mao, and T. T. S. Rappaport, "Directional neighbor discovery in mmwave wireless networks," *Digit. Commun. Netw.*, vol. 7, no. 1, pp. 1–15, 2021.

[37] D. Zhang, P. S. Santhalingam, P. Pathak, and Z. Zheng, "Characterizing interference mitigation techniques in dense 60 ghz mmwave wlans," in *28th Int. Conf. Comput. Commun. Netw. (ICCCN)*, 2019.

[38] K. Hosoya, N. Prasad, K. Ramachandran, N. Orihashi, S. Kishimoto, S. Rangarajan, and K. Maruhashi, "Multiple sector id capture (midc): A novel beamforming technique for 60-ghz band multi-gbps wlan/pan systems," *IEEE Trans. Antennas Propag.*, vol. 63, no. 1, pp. 81–96, 2015.

[39] 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 *11th ACM Conf. Emerg. Netw. Experiments Technol.*, 2015.

[40] S. Dutta, C. N. Barati, D. Ramirez, A. Dhananjay, J. F. Buckwalter, and S. Rangan, "A case for digital beamforming at mmwave," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 756–770, 2020.

[41] F. Gómez-Cuba, T. Zugno, J. Kim, M. Polese, S. Bahk, and M. Zorzi, "Hybrid beamforming in 5g mmwave networks: a full-stack perspective," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2021.

[42] R. J. Mailloux, *Phased array antenna handbook.* Artech house, 2017.

[43] T. Haslwanter, *3D Kinematics.* Springer, 2018.

[44] N. Yoganandan, F. A. Pintar, J. Zhang, and J. L. Baisden, "Physical properties of the human head: Mass, center of gravity and moment of inertia," *J. Biomechanics*, vol. 42, no. 9, pp. 1177–1192, 2009.

[45] K. Shoemake, "Animating rotation with quaternion curves," *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, p. 245–254, 1985.

[46] W. H. Von Aulock, "Properties of phased arrays," *Proc. IRE*, vol. 48, no. 10, pp. 1715–1727, 1960.

[47] T. He and Z. Xiao, "Suboptimal beam search algorithm and codebook design for millimeter-wave communications," *Mobile Netw. Appl.*, vol. 20, no. 1, pp. 86–97, 2015.

[48] I. Ahmed, H. Khammari, A. Shahid, A. Musa, K. S. Kim, E. De Poorter, and I. Moerman, "A survey on hybrid beamforming techniques in 5g: Architecture and system model perspectives," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3060–3097, 2018.

[49] J. Kuffner, "Effective sampling and distance metrics for 3d rigid body path planning," in *IEEE Int. Conf. Robot. and Automat.*, vol. 4, 2004, pp. 3993–3998.

[50] K. Shoemake, "Uniform random rotations," in *Graph. Gems III (IBM Version)*, 1992, pp. 124–132.

[51] J. Song, J. Choi, and D. J. Love, "Codebook design for hybrid beamforming in millimeter wave systems," in *IEEE Int. Conf. Commun. (ICC)*, 2015, pp. 1298–1303.

[52] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[53] D. He, K. Guan, A. Fricke, B. Ai, R. He, Z. Zhong, A. Kasamatsu, I. Hosako, and T. Kürner, "Stochastic channel modeling for kiosk applications in the terahertz band," *IEEE Trans. THz Sci. Technol.*, vol. 7, no. 5, pp. 502–513, 2017.

[54] Z. Muhi-Eldeen, L. Ivrissimtzis, and M. Al-Nuaimi, "Measurements and modelling of cellular interference in local point-to-multipoint distribution systems," *IET Microw., Antennas Propag.*, vol. 3, pp. 250–259(9), March 2009.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3229134

15

[55] J. Sarris and A. R. Nix, "Ricean k-factor measurements in a home and an office environment in the 60 ghz band," in *16th IST Mobile Wireless Commun. Summit*, 2007, pp. 1–5.

[56] M. K. Samimi, S. Sun, and T. S. Rappaport, "Mimo channel modeling and capacity analysis for 5g millimeter-wave wireless systems," in *10th Eur. Conf. Antennas Propag. (EuCAP)*, 2016, pp. 1–5.

[57] A. Alayón Glazunov, A. Razavi, R. Maaskant, and J. Yang, "Semi-analytical model of the rician k-factor," *Radio Sci.*, vol. 55, no. 11, 2020.

[58] X. Chen, S. Zhang, and Q. Li, "A review of mutual coupling in mimo systems," *IEEE Access*, vol. 6, pp. 24 706–24 719, 2018.

[59] H. Singh, H. Sneha, and R. Jha, "Mutual coupling in phased arrays: A review," *Int. J. Antennas Propag.*, vol. 2013, pp. 1–23, 2013.

[60] X. Chen, M. Abdullah, Q. Li, J. Li, A. Zhang, and T. Svensson, "Characterizations of mutual coupling effects on switch-based phased array antennas for 5g millimeter-wave mobile communications," *IEEE Access*, vol. 7, pp. 31 376–31 384, 2019.

[61] P. Yang, T. Q. Quek, J. Chen, C. You, and X. Cao, "Feeling of presence maximization: mmwave-enabled virtual reality meets deep reinforcement learning," *arXiv preprint arXiv:2107.01001*, 2021.

[62] K. W. Cho, M. H. Mazaheri, J. Gummeson, O. Abari, and K. Jamieson, "Mmwall: A reconfigurable metamaterial surface for mmwave networks," in *22nd Int. Workshop Mobile Comput. Syst. Appl.*, ser. HotMobile '21. ACM, 2021, p. 119–125.

Jeroen Famaey is an associate professor at imec and the University of Antwerp, Belgium. He received his M.Sc. degree in Computer Science from Ghent University, Belgium in 2007 and a Ph.D. in Computer Science Engineering from the same university in 2012. He is co-author of over 160 articles published in international peer-reviewed journals and conference proceedings, and 7 granted patents. His research focuses on performance modelling and optimization of wireless networks and embedded computing systems, with a specific interest in sustainable IoT and 6G technologies.

Jakob Struye is a Ph.D. researcher in the field of wireless networking at the IDLab research group (University of Antwerp) and imec research institute, Belgium. He obtained his B.Sc (2015) and M.Sc. (2017) in Computer Science at the University of Antwerp. His current research focuses on leveraging extremely high frequency wireless networks in the millimeter wave bands to improve the performance of truly wireless Virtual and Augmented Reality experiences, and he has experience in applying Artificial Intelligence to time series prediction problems. He authored several publications in these fields. He also participated in the DARPA Spectrum Collaboration Challenge (2019), with his team participating in the finals.

Filip Lemic received his B.Sc. and M.Sc. from the University of Zagreb in 2010 and 2012, and his Ph.D. from the Technische Universität Berlin in 2017. Currently, he is a senior researcher at the i2Cat Foundation. He was a postdoctoral researcher and Marie Curie fellow at the University of Antwerp (2018-22) and Universitat Politècnica de Catalunya (2020-22). He was also affiliated with imec (2018-22), FIWARE Foundation (2018), and Technische Universität Berlin (2012-17). He was a visiting researcher at the University of California at Berkeley (2015-16) and Shanghai Jiao Tong University (2019-20). He co-authored more than 60 peer-reviewed research articles and was involved in various international research projects, notably EU MSCA ScaLeITN, EU EVARILOS, EU eWine, NIST's PerfLoc, and UC Berkeley's beyond-5G.