Multi-Armed Bandit Dynamic Beam Zooming for mmWave Alignment and Tracking

Nathan Blinn, *Student Member, IEEE* and Matthieu Bloch, *Senior Member, IEEE*
Emails: nblinn6@gatech.edu and matthieu.bloch@coe.gatech.edu

***Abstract*—We propose an Integrated Sensing and Communication (ISAC) algorithm that exploits the structure of a hierarchical codebook of beamforming vectors using a best-arm identification Multi-Armed Bandit (MAB) approach for initial alignment and tracking of a Mobile Entity (ME). The algorithm, called Dynamic Beam Zooming (DBZ), performs beam adjustments that mitigate the severe outages associated with wireless mmWave systems and allow for adaptive control of the parameters governing communications. We analyze the sample complexity of DBZ and use it to inform how the algorithm adapts to the non-stationary MAB statistics based on ME motion and Signal-to-Noise Ratio (SNR). We perform extensive simulations to validate the approach and demonstrate that DBZ is competitive against existing Bayesian algorithms, without requiring channel multi-path or fading knowledge. In particular, DBZ outperforms other low-complexity algorithms in the low SNR regime. We also illustrate the efficacy of DBZ in standardized rural and urban scenarios using NYU Sim.**

***Index Terms*—5G, 6G, millimeter-Wave, MIMO, Beamforming, Multi-Armed Bandits.**

I. Introduction

Communication in the millimeter-Wave (mmWave) spectrum (30 GHz to 300 GHz) is envisioned as a key enabler of ultra-high-speed data delivery with low latency for next generation wireless systems [1]. The severe path loss inherently associated with mmWave frequencies, however, creates unique engineering challenges. First, compensating for the path loss requires transceivers to combine massive Multiple-Input Multiple-Output (MIMO) arrays to form highly focused, narrow beams [2] together with Hybrid Analog-Digital (HAD) architectures to reduce the otherwise impractical number of associated Radio Frequency (RF) paths [3]. Second, ensuring persistent and reliable communication between entities requires efficient beam refinement and management to initiate alignment and track ME movement over time [4]. Beam alignment and tracking can be viewed as *sensing* tasks, leveraging approaches in radar [5], that support a *communication* task. While the two tasks could be independently addressed, joint designs within the framework of ISAC offer opportunities for enabling emerging applications [6] and efficiently utilize increasingly congested wireless resources and constrained hardware [7].

A. Related Works

The 5G standard currently only offers basic support for mmWave beam alignment and tracking in the form of an exhaustive search for beam directions [8, Section 4]. Consequently, several classes of ISAC algorithms for mmWave beam alignment and tracking have been investigated, each offering different complexity-measurement-performance tradeoffs. The classes are summarized in Table I and discussed next.

TABLE I: Comparison of Algorithms with ISAC Defining Features.

Reference	Alignment Complexity	Alignment Accuracy	Motion Adaptive	CSI Adaptive	Computational Overhead
KF [9]	N/A	✗	✓	✗	Low
RL [10], [11]	High	✗	✓	✓	Low
ABP [12]	High	✗	✓	✓	Low
HPM [13]	Low	✓	✗	✗	High
2PHTS [14]	Low	✓	✗	✓	High
HBA [15]	Low	✗	✗	✓	Low
HOSUB [16]	Fixed	✗	✗	✓	Low
DL-IA [17]	Fixed	✓	✗	✗	High*
DL [18]	N/A	✗	✓	✗	High*
PF [19]	N/A	✗	✓	✗	High
ABT [20]	Low	✓	✓	✗	High
Present Work	Med-Low	✓	✓	✓	Med-Low

* High computational overhead pending hardware implementation.

A first class of algorithms uses Bayesian decision-making and leverages hierarchical beamforming codebooks [21]. In particular, [13] proposes the Hierarchical Posterior Matching (HPM) algorithm that exploits Channel State Information (CSI) and measurements to update the posterior probabilities of the incoming beam direction, choosing increasingly narrower beams as the posteriors identify more precisely the likely beam direction. The approach of HPM has also been recently extended to track MEs [22].

A second class of algorithms selects beamformer weights using a Deep Neural Network (DNN) [17] instead of relying on a predefined hierarchical codebook. Numerical results show that the performance of this "codebook-free" approach without

CSI matches the performance of HPM with full CSI. DNN approaches, however, may take a significant number of samples to correctly point a beam. For example, the deep reinforcement learning algorithm in [23] takes about 10^5 samples to converge at runtime.

A third class of algorithms attempts to circumvent the computational complexity incurred by Bayesian and DNN approaches using Compressed Sensing (CS) techniques. The idea is to exploit the sparsity associated with mmWave channels [2] to quickly identify the direction of incoming signal. To infer user location, the approach in [24], [25] is to generate random peaks in multiple beam patterns to quickly identify the optimal combinations to form beamforming weights.

Particularly relevant to the present work, a fourth class of algorithms exploits the conceptual analogy between beam steering and arm play in a Multi-Armed Bandit (MAB) problem to lower complexity without sacrificing performance. In brief, every beam direction, which corresponds to a set of phase shifts applied to array elements, may be viewed as an arm to pull in a MAB algorithm and the Reference Signal Received Power (RSRP) acquired with every choice of beam direction provides a reward that may be exploited by a MAB exploration strategy. Experimental results in [26] show that the alignment of two users exhibits an approximate unimodal structure [27], [28] that can be efficiently exploited. Few reward structures are perfectly unimodal and the algorithm in [26], [27] may get stuck in local maxima. To address this, [16] adapts Optimal Sampling for Unimodal Bandits (OSUB) [27] for use with a hierarchical codebook; numerical simulations show a substantial reduction in the number of samples required, with robustness to multi-path effects but no theoretical guarantees. The Hierarchical Beam Alignment (HBA) algorithm [15] adapts the X-arm bandit algorithm [29], but only indirectly exploits the hierarchical structure since pencil beams acquire reward signals. The Two Phase Heteroscedastic Track-and-Stop (2PHTS) algorithm [14] uses grouped sums of arms as "super-arms," which are broader beams, to create a two-level hierarchical beamformer and adopts the Track-and-Stop (TAS) framework of [30], [31]. With respect to track tasks, MAB algorithms in the regret setting provide a low-overhead approach, where for instance [10] chooses arms close to the empirical best to play at each round.

A final class of algorithms uses tools from adaptive control for Angle of Arrival (AoA) and/or Angle of Departure (AoD) estimation over time. [9] estimates the fading coefficient along with the angles using an Extended Kalman Filter (EKF) for a low computational overhead approach to motion compensation, but no initial alignment. Beamwidth control over time ultimately prevents large outages associated with mmWave channels during instances of misalignment. For instance, [19], [32] both adapt Particle Filter (PF)s to make dynamic adjustments to the beamwidth over time, at the expense of high computational overhead.

B. Contributions and Outline

The main contributions of the present work are as follows:

- We present Dynamic Beam Zooming (DBZ), an ISAC algorithm for mmWave beam alignment and tracking that offers high alignment accuracy and automatic adaptation to changes in CSI and motion while preserving a relatively low complexity. In particular, as summarized in Table I, DBZ strikes competitive performance against Bayesian methods exploiting full channel knowledge and ME motion such as [13], [20].
- We show that DBZ is able to operate in the low SNR regime, avoiding intrinsic numerical issues of competing approaches [14].[1]
- We guarantee the accuracy of the initial alignment phase by showing that DBZ is δ-probably approximately correct (PAC) and derive a closed-form expression for the sample complexity. We also use the sample complexity to inform how the algorithm adapts to the time-varying statistics.
- We provide extensive simulations, including realistic environments from NYU Sim [33], [34].

The remainder of the document is organized as follows. In Section II, we introduce the system model used for our ISAC scenario. In Section III we describe the hierarchical codebooks used for beam alignment and tracking. In Section IV, we introduce DBZ, which uses a MAB best arm identification framework to quickly align and adapts to the ME motion over time. In Section V, we develop closed-form expressions of the sample complexity that inform the choice of parameters in the DBZ algorithm. In Section VI, we present extensive numerical simulations demonstrating the performance DBZ performance across a wide range of scenarios.

II. System Model

A. System Model

At each discrete time step $n \in \mathbb{N}$, a Base Station (BS) transmits a Synchronization Signal (SS)/Reference Signal (RS), $\mathbf{s} \in \mathbb{C}^{Q\times 1}$, consisting of Q samples at finer time or frequency granularity. Each signal consists of time-frequency Resource Elements (RE)s across multiple Orthogonal Frequency Division Multiplexing (OFDM) symbols similar to that of the Synchronization Signal Block (SSB) or CSI-RS used in the current 5G standard [35, 7.4]. Each transmission, $\mathbf{s}$, has a cell identification number that is unique to one BS, where $\mathbf{s}^H\mathbf{s} = 1$. We use a Uniform Linear Array (ULA) of M elements to transmit signals over which we apply a HAD beamforming vector to electronically steer each transmission of $\mathbf{s}$. The HAD beamforming vectors belong to a beamforming codebook, $\mathcal{F}$, that consists of analog phase shifts for M antenna elements with N_{RF} RF chains, $\mathbf{F}_{\text{RF}} \in \mathbb{C}^{M\times N_{\text{RF}}}$, and a digitally applied baseband precoder, $\mathbf{F}_{\text{BB}} \in \mathbb{C}^{N_{\text{RF}}\times N_S}$, for each RF chain to feed N_S datastreams. We denote the combined beamforming vector for a single datastream, u, as $\mathbf{f} = \mathbf{F}_{\text{RF}}[\mathbf{F}_{\text{BB}}]_u$ ($\mathbf{f}^H\mathbf{f} = 1$), where $[\mathbf{F}_{\text{BB}}]_u$ is the u^{th} column of $\mathbf{F}_{\text{BB}}$. The beamforming pattern for each of the vectors, $\mathbf{f} \in \mathcal{F}$, has a unique pointing angle, $\bar{\phi} \in \Phi \triangleq [\phi_{\min}, \phi_{\max}]$, with $\bar{\phi}$ evenly spaced across a predefined range Φ and with each pattern having an equivalent *beamwidth*, ϕ_{bw}, and *gain*,

[1] 2PHTS uses an approximation of the actual channel model. The empirical mean may become negative in the low-SNR regime, causing numerical issues in the computation of the relative entropy with the TAS baseline algorithm.

g. We require steering the beamforming pattern in the angular direction of the receiving ME, $\theta_k(n)$.

B. Channel and Kinematic Models

We assume that the receiving ME forms a single beam, such that it is always directed at the transmitting BS or omni-directional.[2] We consider only a single subcarrier to allow for the narrowband channel representation [37, Eq. (7)],

$$\mathbf{h}(n) = \sum_{k=1}^{K} \alpha_k(n)\mathbf{a}^H(\theta_k(n)), \tag{1}$$

where

$$\mathbf{a}(\theta) \triangleq \left[e^{-j\frac{M-1}{2}\frac{2\pi d\cos(\theta)}{\lambda}} \quad \dots \quad e^{j\frac{M-1}{2}\frac{2\pi d\cos(\theta)}{\lambda}}\right]^T \tag{2}$$

is the array response for a ULA, $j^2 \triangleq -1$, and d and λ are the array element spacing and wavelength, respectively. For each path, $\alpha_k(n) \in \mathbb{C}$ represents the complex gain caused by large and small scale fading.[3] We assume $\theta_1(n)$ is the dominant path in a Line of Sight (LOS) scenario with the receiving ME. The received signal takes the form

$$z(n) = \mathbf{h}(n)\mathbf{f}\mathbf{s}^T\mathbf{s}^* + \mathbf{v}^T(n)\mathbf{s}^*(n) = \mathbf{h}(n)\mathbf{f} + v(n) \tag{3}$$

where $\mathbf{v}(n) \sim \mathcal{CN}(0, \sigma_v^2\mathbf{I})$. The RSRP measurement is

$$y(n) = |z(n)|^2 . \tag{4}$$

The receiving ME regularly communicates control data or measurements to the BS advising beamforming vector selection, similar to the 5G/New Radio (NR) standard [8, Section 6] [38, Section 5.6.1] [39, Section 5].

Between discrete time steps n and $n+1$, spaced τ seconds apart, the BS and ME experience relative motion according to a Discrete White Noise Acceleration (DWNA) motion model [40, Chapter 6.3.2],

$$\theta(n) = \theta(n-1) + \tau\dot{\theta}(n-1) + \frac{\tau^2}{2}u(n-1), \tag{5}$$

$$\dot{\theta}(n) = \dot{\theta}(n-1) + \tau u(n-1), \tag{6}$$

where $\dot{\theta}(n)$ is the angular velocity and $u(n) \sim \mathcal{N}(0, \sigma_u^2)$. Standard deviation of the acceleration, σ_u, governs the severity of the motion between time steps. We simulate the operation of DBZ with the DWNA model in Section VI for various values of σ_u, governing the severity of the motion.

C. Alignment Problem

The relative motion requires adjustments to the beamforming vector to maintain *alignment*. We define alignment as the state in which we choose the beamforming vector, $\mathbf{f}^*(n)$ at time step n, such that[4]

$$\mathbf{f}^*(n) = \underset{\mathbf{f}\in\mathcal{F}}{\operatorname{argmin}} \|\mathbf{f} - \mathbf{a}(\theta(n))\|. \tag{7}$$

For a beamforming vector $\mathbf{f}^*(n)$ pointed towards the angle $\bar{\phi}$, we define N^{a} as the maximum number of consecutive time steps during which $\theta(n) \in \mathcal{R} = [\bar{\phi} - \phi_{\text{bw}}/2, \bar{\phi} + \phi_{\text{bw}}/2]$, where $\mathcal{R}$ represents the coverage region of the beamforming vector. To achieve (7) with a probability of at least $1-\delta$, we employ a MAB best-arm identification strategy, based on [41], to select a beamforming vector. Following the beamforming vector selection, we monitor the RSRP measurements over time, utilizing the same signals used for communication, where abrupt changes in power serve as indicators of misalignment.

III. Hierarchical Codebook and Structure

Our work exploits a HAD *hierarchical* codebook $\mathcal{F}^H$ adapted from [3]. As illustrated in Fig. 1, our construction is as follows.

- The codebook consists of H levels, each level $h \in \{1, \dots, H\}$ corresponding to beams with beamwidth $\phi_{\text{bw},h}$;
- Each beam at level h is split into three non-overlapping narrower beams at level $h+1$ so that $\phi_{\text{bw},h} = 3\phi_{\text{bw},h+1}$;
- The gain at each level h is $g_h = g^{h-H+1}$;
- Each beam at level h is also associated to a broader beam pointed in the same angle for all $h < H$.

Mathematically, this means there are exactly $I = 3^{H-1}|\mathcal{I}_1|$ beamforming vectors at each level h with pointing angles

$$\bar{\phi}_{h,i} = \phi_{\min} + \frac{\phi_{\text{bw},H}}{2} + (i-1)\phi_{\text{bw},H}, \quad i \in \{1, \dots, I\}. \tag{8}$$

Each beam identified by (h, i) aggregates a unique set of three non-overlapping patterns with indices $(h+1, j)$ with $j \in \{i - 3^{H-h}, i, i + 3^{H-h}\}$. Each beam (h, i) has a corresponding beam $(h-1, i)$ with the same pointing angle. This construction

[4]To ensure that our codebook contains a beamforming vector that points in the direction $\theta(n)$ in our simulations, we wrap the angle $\theta(n)$ to constrain it to Φ.

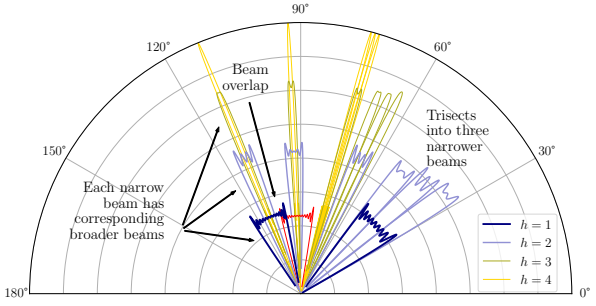

Fig. 1: Example beamforming patterns for the hierarchical codebook.

[2]While this model abstracts away the joint alignment process at the ME and BS, it still captures the essence of the problem and has been widely adopted [13]–[15], [36].

[3]For each path, $\alpha_{k,I}(n) + j\alpha_{k,Q}(n) = \alpha_k(n) \in \mathbb{C}$, we specify later that each component amplitude changes over time according to $\alpha_{k,I}(n+1) = \rho\alpha_{k,I}(n) + \omega_I(n)$, where $\omega_I(n) + j\omega_Q(n) = \omega(n) \sim \mathcal{CN}(0, (1-\rho^2))$ [9] or by the Rician AR-1 channel model in [13].

allows one to quickly "zoom in" from a beam (h,i), narrowing the beamwidth[5] by aggregating the beams in the set

$$\mathcal{Z}_{h,i} \triangleq \begin{cases} \{(h+1,i),(h+1,i\pm 3^{H-h})\} & \text{if } h<H, \\ \{(H,i)\} & \text{if } h=H. \end{cases} \quad (9)$$

In case of misalignment, the codebook allows one to zoom out from (h,i), to $(h-1,i)$ without changing the pointing angle.

A. Codebook Characteristics

We briefly discuss codebook depth, branching factors, and design methodology in how they impact DBZ alignment accuracy, computational complexity, and robustness in dynamic environments. DBZ is agnostic to codebook depth (determined by H). We show in Section V-F that the algorithm parameters may be calibrated to support codebook designs with different choices of beamwidths at various depth levels to most efficiently ensure alignment accuracy with an ME. The branching factor is the main restricting codebook design characteristic for DBZ. To ensure alignment accuracy when broadening a beam and zooming out, each broad beam must be divisible into an odd-numbered quantity of beams to preserve the pointing angle of the previous narrow beam. We exclusively use a branching factor of 3 for this work, but DBZ easily adopts any codebook with an odd-numbered branching factor. Increased depth and larger branching factors contribute to higher sample complexity due to the increased total number of beamforming vectors. However, the increase in depth or branching factor provides more precise alignment with the finer resolution of the search space, Φ. On the other hand, DBZ benefits from shallower codebooks in the case of highly sporadic motion to more quickly adapt to the ME position and maintain alignment accuracy. DBZ may adapt other HAD codebook construction methodologies outside of [3] without compromising performance. Additional logic for DBZ allows easy extension to use adaptive constructed codebooks, as in [42], to further reduce training overhead for multiple ME.

B. Induced Mean Reward Structure

For any beamforming vector $\mathbf{f}_{h,i} \in \mathcal{F}^H$, $|z_{h,i}(n)|^2$ is a $\sigma_v^2/2$-scaled non-central chi-squared random variable with two degrees of freedom, and has non-centrality parameter, $2\zeta_{h,i}(n)/\sigma_v^2$, where $\zeta_{h,i}(n) = |\mathbf{h}(n)\mathbf{f}_{h,i}|^2$. We define the mean-reward function generated by the RSRP measurements (4) of the channel as

$$\mu_{h,i}(n) \triangleq \mathbb{E}\Big(|\mathbf{h}(n)\mathbf{f}_{h,i} + v(n)|^2\Big). \quad (10)$$

Using the hierarchical codebook results in an induced structure of the mean rewards, we make two assumptions.

Assumption 1. *For each h, at any time step, n, there exists a unique beamforming vector $\mathbf{f}_{h,i^*}$ such that*

$$\mu_h^*(n) = \mu_{h,i_h^*}(n) = \max_{i\in\{1,\dots,I\}} \mu_{h,i}(n). \quad (11)$$

[5]In our simulations showing initial alignment performance comparison, we adopt the binary codebook from [3], [13].

We define an ϵ-optimal arm as $(H,i^\epsilon) \in \{(H,i) : \mu_{H,i}(n) + \epsilon \geqslant \mu_H^*(n)\}$. By definition, (H,i^*) is ϵ-optimal.

Assumption 2. (Unimodality) *For all n, if $\mu_{H,i^\epsilon}(n) + \epsilon \geqslant \mu_H^*(n)$ then there exist paths $((1,i_1),(2,i_2),\dots,(H-1,i_{H-1}),(H,i^\epsilon))$ through the tree graph defining the codebook where*

$$\mu_{H,i^\epsilon}(n) > \mu_{H-1,i_{H-1}}(n) > \cdots > \mu_{2,i_2}(n) > \mu_{1,i_1}(n) \quad (12)$$

The sparsity and high path loss attenuation associated with mmWave propagation [43] suggest that Assumptions 1 and 2 hold in most situations. We denote the difference between mean rewards at a particular level h as

$$\Delta_{h,i}(n) \triangleq \begin{cases} \mu_h^*(n) - \mu_{h,i}(n) & \text{if } i \neq i^*, \\ \mu_h^*(n) - \max_{i\neq i^*} \mu_{h,i}(n) & \text{if } i = i^*. \end{cases} \quad (13)$$

Our analysis and discussion in Section IV-E emphasizes that the spacing between mean rewards, $\Delta_{h,i}(n)$, significantly contributes to overall sample complexity. In particular, broader beams will have smaller values of $\Delta_{h,i}(n)$, and therefore higher sample complexity. Section IV-F shows how to configure DBZ such that we play certain levels and optimize the trade off of sample complexity and number of beamforming vectors played. From our codebook construction, for any ϵ-optimal arm, there exists a path $\{(h,i_h)\}|_{h=1}^H$ such that

$$\frac{\mu_{H,i^\epsilon}(n)}{\mu_{H-1,i_{H-1}}(n)} = \frac{\mu_{H-1,i_{H-1}}(n)}{\mu_{H-2,i_{H-2}}(n)} = \cdots = \frac{\mu_{2,i_2}(n)}{\mu_{1,i_1}(n)} = g. \quad (14)$$

From $\mu_{H,i^\epsilon}(n)+\epsilon \geqslant \mu_H^*(n)$, (14) ensures that $\mu_{H,i^\epsilon}(n)+\epsilon \geqslant g\mu_{H-1}^*(n)$, from which we obtain $\mu_{h,i_h}(n) + \epsilon_h \geqslant \mu_h^*(n)$, where $\epsilon_h \triangleq g^{-(H-h)}\epsilon$. If the average reward corresponding to beamforming vector $\mathbf{f}_{h,i}$ meets the criteria of $\mu_{h,i}(n) \geqslant \mu_h^*(n) + \epsilon_h$ then it is ϵ_h-optimal. We relate the relative cost to spectral efficiency to ϵ in Section V. In our model (5), the mean rewards are non-stationary, causing the unique maximum mean-reward, $\mu_H^*(n)$, to change over time. The next section introduces our algorithm, DBZ, that dynamically adjusts the beamwidth used for communication by selecting beamforming vectors under certain *zoom-in* and *zoom-out* criteria, based on MAB best arm identification and power threshold, respectively, to maintain alignment with the ME.

IV. Algorithm: Dynamic Beam Zooming

DBZ uses the hierarchical codebook described in Section III and efficiently exploits the induced dynamic reward structure. DBZ exploits the representation of each beamforming vector $\mathbf{f}_{h,i}$ as a vertex (h,i) in a tree and uses a best arm identification MAB framework [41] and power threshold to dynamically navigate the tree and maintain alignment with the ME. Informally, the algorithm operates as illustrated in Fig. 2 to show example of traversing the graph vertices for beam refinement. Vertices with an asterisk indicate the beam used to communicate and the triangle moving along the bottom represents an ME. The leafs at the bottom of each tree represent the narrowest beams.

- Steps 1 and 2: to initially align, we identify with probability $1-\delta$ the beamforming vector $\mathbf{f}_{H,i_H^*}$ that most closely matches the ULA response to $\theta(n)$ according to (7) within N_h^{a} time steps. This is achieved with MAB algorithms that, at levels h, play beamforming vectors $\mathbf{f}_{h,i}$ viewed as arms in a MAB best-arm identification fixed confidence setting. The chosen arm, corresponding to a narrower beam, is used for increasing the rate at which we communicate data. We then put the chosen arm's zoom-in indices $\mathcal{Z}_{h,i}$ in (9) in contention to play a subsequent MAB game, to continue to refine the communication beamwidth.
- Steps 2 to 3: DBZ detects beam misalignment by the RSRP failing to meet a power threshold, and "zooms out", adjusting the set of active vertices.
- Step 4: the broader beam is adjusted to realign.
- Step 5: the beam is correctly re-adjusted to the narrowest width.

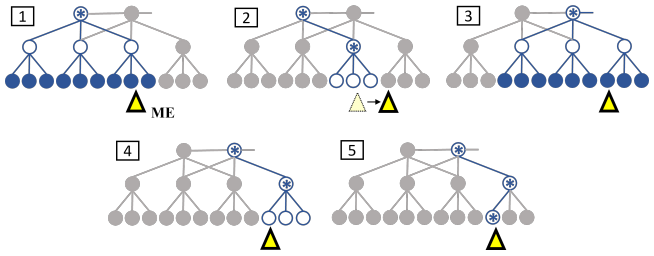

Fig. 2: Illustration of beamforming vector selection in DBZ over time.

A. Baseline Framework

DBZ proceeds mathematically using the Lower-Upper Confidence Bound (LUCB) best-arm identification framework [41], [44] for each MAB game. LUCB uses empirical statistics derived from the sample reward values (in this case RSRP) that represent the estimation and uncertainty on the mean rewards. Due to the non-stationary rewards from the ME motion, we only consider a finite set of size η consisting of the most recent samples to compute the LUCB statistics. We refer to the finite set as the *sample window* [45]. At each time step n, we select a specific beamforming vector (h,i) whose indices are stored as $S(n)$ and observe the corresponding reward, $y(n)$. The mean rewards, $\mu_{h,i}(n)$ (10), at time step n are estimated by the empirical mean, using the η most recent samples according to

$$\hat{\mu}_{h,i}(\eta,n)=\frac{1}{N_{h,i}(\eta,n)}\sum_{p=\max\{1,n-\eta\}}^{n} y(p)\mathbb{1}\{S(p)=(h,i)\}\,, \tag{15}$$

where[6]

$$N_{h,i}(\eta,n)=\sum_{p=\max\{1,n-\eta\}}^{n}\mathbb{1}\{S(p)=(h,i)\}\,. \tag{16}$$

[6]For the indicator function, $\mathbb{1}\{S(n)=(h,i)\}=1$ when $S(n)=(h,i)$ and 0 otherwise.

To allow further generalization later on, we let $\mathcal{I}_h$ denote the set of arms in contention at level h, noting that $|\mathcal{I}_h|=3$ for all levels except $h=1$, and let $\mathcal{I}^H=\sum_h|\mathcal{I}_h|$. Pictorially, each level's active vertices in Fig. 2 represent arms in $\mathcal{I}_h$. For constants $B,C\geqslant 1$ to be chosen later, we use a confidence term, empirical observation variance estimate, and exploration rate

$$D_{h,i}(\eta,n)\triangleq\sqrt{\frac{4B\hat{\nu}_{h,i}^2(\eta,n)\beta(\eta,n,\delta)}{N_{h,i}(\eta,n)}}+\frac{2\sqrt{2BC}\beta(\eta,n,\delta)}{N_{h,i}(\eta,n)-1}, \tag{17}$$

$$\hat{\nu}_{h,i}^2(\eta,n)=\sum_{p=\max\{1,n-\eta\}}^{n}\frac{(y(p)-\hat{\mu}_{h,i}(\eta,p))^2\mathbb{1}\{S(n)=(h,i)\}}{N_{h,i}(\eta,n)}, \tag{18}$$

$$\beta(\eta,n,\delta)\triangleq\log\left(15\mathcal{I}^H(\min\{n,\eta\})^4/(2\delta)\right), \tag{19}$$

respectively, in the Upper-Confidence Bound (UCB) and Lower-Confidence Bound (LCB) terms

$$U_{h,i}(\eta,n)=\hat{\mu}_{h,i}(\eta,n)+D_{h,i}(\eta,n), \tag{20}$$

$$L_{h,i}(\eta,n)=\hat{\mu}_{h,i}(\eta,n)-D_{h,i}(\eta,n), \tag{21}$$

respectively. The terms, (20) and (21), capture the best and worst performance, respectively, of a beamforming vector, that we use to define the *gap* for each arm,

$$G_{h,i}(\eta,n)=\max_{j\neq i}U_{h,j}(\eta,n-1)-L_{h,i}(\eta,n-1), \tag{22}$$

and the indices

$$\gamma(n)=\operatorname*{argmin}_{i:(h,i)\in\mathcal{I}_h}G_{h,i}(\eta,n), \tag{23}$$

$$u(n)=\operatorname*{argmax}_{i:(h,i)\in\mathcal{I}_h,i\neq\gamma(n)}U_{h,i}(\eta,n-1). \tag{24}$$

We sample a beamforming vector $\mathbf{f}_{S(n)}$ with index tuple

$$S(n)\triangleq\operatorname*{argmax}_{(h,i):i\in\{\gamma(n),u(n)\}}D_{h,i}(\eta,n-1), \tag{25}$$

or all $(h,i)\in\mathcal{I}_h$ in round-robin fashion to first initialize a new level. The individual MAB games are independent across levels, where "zooming in" to the next level is governed by *termination* at the current level. Termination and zooming in at a particular level h occurs when the gap term for $\gamma(n)$ first satisfies:

$$G_{h,\gamma(n)}(n)=U_{h,u(n)}(\eta,n-1)-L_{h,\gamma(n)}(\eta,n-1)<\epsilon_h, \tag{26}$$

at which point, we choose $(h,\gamma(n))$ for communication, and store it as (h,i^c). Intuitively, $L_{h,\gamma(n)}(\eta,n-1)$ is the worst performance of the estimated best beamforming vector and $U_{h,u(n)}(\eta,n-1)$ is the best performance of the runner-up beamforming vector. We show in our analysis in Section V that with (26) we make a correct selection beam, i.e., $(h,\gamma(n))=(h,i^*)$, of a beamforming vector at level h with probability at least $1-\delta$. We next show how this LUCB mathematical framework is used to facilitate DBZ.

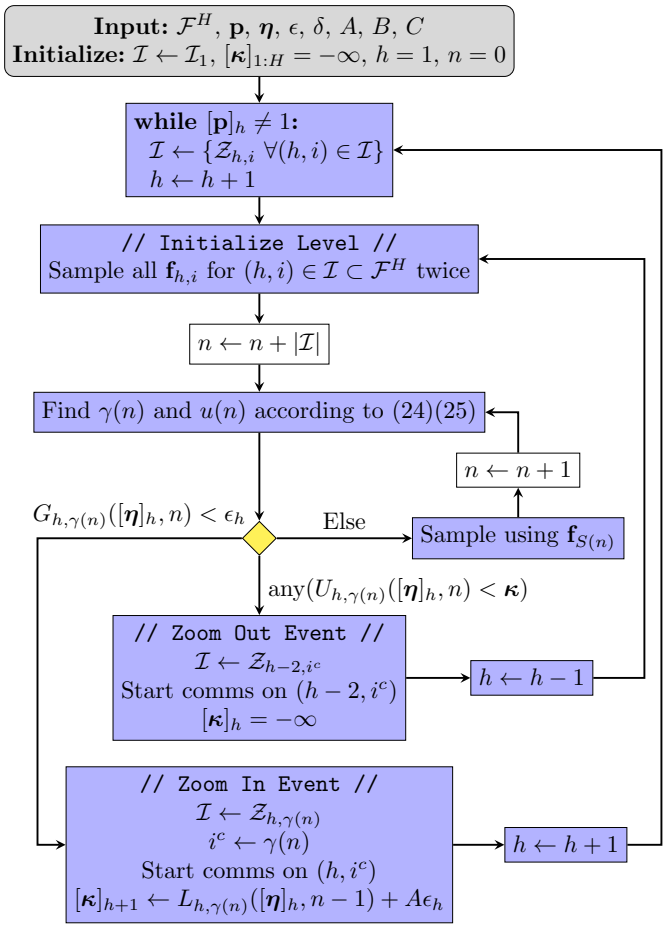

Fig. 3: Flowchart of DBZ algorithm.

B. DBZ Algorithm

Fig. 3 shows how the DBZ algorithm proceeds with zooming in or zooming out using the baseline LUCB MAB framework. Each level has a different beamwidth, $\phi_{\text{bw},h}$, which implies a different alignment time. We use a vector of hyperparameters, $\boldsymbol{\eta} \in \mathbb{N}^H$, whose h^{th} element is the sampling window length used at level h. For now, assume $\mathbf{p} = [1, \ldots, 1]$. DBZ initially samples from a fixed set of beamforming vectors, $\mathcal{I}_1$.[7] $\mathcal{I}_1$ possesses only the broadest beam patterns that are non-overlapping and perfectly cover Φ. DBZ checks the termination criteria in (26) to determine zooming in at the western path of the decision diamond in Fig. 3. Once the algorithm terminates at the initial level, $h = 1$, we begin to communicate using the chosen beamforming vector $(1, \gamma(n))$, stored as (h, i^c). DBZ continues to play MAB games at subsequent levels, choosing $(h, \gamma(n))$ upon termination at level h, and refines the communication beam with the subsequent MAB games at $h > 1$. As shown in Fig. 2, this operation continues until DBZ terminates with a narrowest beam at level H. Fig. 3 shows DBZ loops back to initialize the next level after zooming in. Conversely, following the southern path of the decision diamond in Fig. 3, we control "zooming out" to a previous level with a wider beam by establishing a vector of hyperparameters, $\boldsymbol{\kappa}$, whose elements are the threshold RSRP after termination and zooming in at each level,

$$[\boldsymbol{\kappa}]_{h+1} \triangleq L_{h,\gamma(n)}(\eta, n-1) + A\epsilon_h, \tag{27}$$

for $A \leqslant 1$ to be chosen later. If at any time step

$$U_{h,\gamma(n)}(\eta, n-1) < [\boldsymbol{\kappa}]_{h'}, \tag{28}$$

for any $h' \leqslant h$, DBZ zooms out choosing $(h-1, i^c)$ as the new communication beam and loops back on the flowchart re-initializing all arms in $\mathcal{Z}_{h-2,i}$ that are now in contention.[8] In the case of zooming out at $h = 2$, we reset and store $\mathcal{I}_1$ as $\mathcal{I}$. The intuition for our choice of of adaptive threshold in (27) is that we base it on the worst performance of a previous level's ($h' < h$) best-performing beamforming vector, $L_{h',\gamma(n)}(\eta, n-1)$. If the best performance of the current level's best-performing beamforming vector, $U_{h,\gamma(n)}(\eta, n-1)$, does not exceed the threshold, one concludes misalignment and zooms out. These discrete decision points for beam transitions allow the transmission of control information between the BS and ME to adjust the corresponding rate and beamforming vector [8], [38]. In the case of neither zooming in or out, we take the eastern path of the decision diamond and sample beamforming vector corresponding to $S(n)$.

Playing each level reduces the number of arms considered overall, but *naively* exploits the hierarchical codebook. Certain levels of the codebook are more beneficial to play than others based on the number of beams eliminated per number of samples required. Consequently, one might benefit from skipping some levels at the expense of contending more arms in a best-arm identification MAB game. We characterize the strategy used to navigate the codebook levels by a pruning vector of hyperparameters $\mathbf{p}$ with elements $[\mathbf{p}]_h \in \{0, 1\}$. Specifically, assume that at level $h-1$, the beamforming vectors corresponding to the vertices in $\mathcal{I}_{h-1}$ are used in the best-arm identification MAB game to take samples as RSRP measurements. Upon termination, if $[\mathbf{p}]_h = 1$, the children vertices' beamforming vectors of the arm chosen at level $h-1$ are played in the next level h. If $[\mathbf{p}]_h = 0$, we bypass level h and, pending $[\mathbf{p}]_{h+1} = 1$, put all descendant arms in contention (See block prior to level initialization in Fig. 3). If $[\mathbf{p}]_{h+1} = 0$, we bypass this level, and so on. As an example, the size of a set of beamforming vectors after skipping one level is $|\mathcal{I}| = 9$ for a ternary tree, or $|\mathcal{I}| = 4$ for a binary tree. Note that $[\mathbf{p}]_H$ must be set to 1 because we require a choice of one of the narrowest beamforming patterns. Our simulations in Section VI show that many of the hyperparameters may be generically set over a broad range of channel conditions and maintain performance.

V. Analysis

For initial alignment, DBZ adapts the fixed-confidence best arm identification framework in [41], in which the algorithm requires choosing the correct beam with high probability. DBZ requires accurate estimation of the mean reward values in (10)

[7] Note that in Fig. 3 we store $\mathcal{I}_1$, or $\mathcal{I}_h$, as $\mathcal{I}$.

[8] The logical test $\texttt{any}(\cdot)$ (southern path of the decision diamond in Fig. 3) returns Boolean `True` if any element in a logic vector returns `True`.

for $(h,i) \in \mathcal{I}_h$ while maintaining alignment. For these two requirements, we define the events,

$$\mathcal{B}_h \triangleq \big\{\forall (h,i) \in \mathcal{I}_h, \forall n > 2\,|\mathcal{I}_h|\,, \tag{29}$$

$$|\hat{\mu}_{h,i}(\eta,n) - \mu_{h,i}(n)| < D_{h,i}(\eta,n)\big\}, \tag{30}$$

and

$$\mathcal{A}_h \triangleq \{\exists (h,i) \in \mathcal{I}_h : \theta(n) \in \mathcal{R}_{h,i}\}\,, \tag{31}$$

for a beamforming vector $\mathbf{f}_{h,i}$ whose beampattern is pointed toward angle $\bar{\phi}_i$ and has beamwidth $\phi_{\text{bw},h}$. As a reminder, $\mathcal{R}_{h,i} = [\bar{\phi}_{h,i} - \phi_{\text{bw,h}}/2, \bar{\phi}_{h,i} + \phi_{\text{bw,h}}/2]$ is the coverage region of the beam pattern corresponding to (h,i). Under events (29) and (31) $\forall h \in \{1,\dots,H\}$, we show that DBZ zooms in to choose an ϵ-optimal beamforming vector with probability at least $1-\delta$. In the case the ME is out of alignment, $\mathcal{A}_h^c$, we zoom out to mitigate severe outages, which keeps the ME aligned in the wider beam. We confirm a zoom-out action from (h,i) to (h',i), where $h' < h$, as being *correct* when the mean reward of the broader beam $\mu_{h',i}(n) > \mu_{h,i}(n)$. We first prove in Section V-A why $\mathcal{B}_h$ holds with high probability, ensuring $\mu_{h,i}(n)$ is well estimated. In Section V-B, based on our kinematic motion model (5), we analyze the maximum sample window lengths that may be chosen at each level, $[\boldsymbol{\eta}]_h$, such that $\mathcal{A}_h$ holds (aligned) with high probability. We then show the correctness of our decision criteria for zooming in (26) and zooming out (28) to adjust to the dynamically changing mean reward values in Section V-D. Finally, in Section V-E we develop a means to calculate the sample complexity required to zoom in with respect to the spacings of mean rewards (13) and choice of ϵ. We use the sample complexity to select the hyperparameters corresponding to sample window lengths, $\boldsymbol{\eta}$, pruning vector, $\mathbf{p}$.

A. Confidence

We first provide two supporting lemmas that lay the foundation for ensuring that DBZ correctly zooms in and zooms out with high confidence using the mathematical components in Section IV-A. We use the lemmas to show $\mu_{h,i}$ is well estimated if the event $\mathcal{B}_h$ occurs with high probability during DBZ for all levels. We conclude that with probability at least $1-\delta$, the true mean reward satisfies $\mu_{h,i}(\eta,n) \in [L_{h,i}(\eta,n), U_{h,i}(\eta,n)]$ during execution of DBZ. We let $n_{h,i} \in \mathbb{N}$ denote the time steps at which arm (h,i) is sampled.

Lemma 1. *For the sequence of observations, $\{y(n_{h,i}) : n_{h,i} \geqslant 2\}$, which follow a $\sigma_v^2/2$-scaled non-central chi-squared distribution,*

$$\mathbb{P}(|\hat{\mu}_{h,i}(\eta,n) - \mu_{h,i}(n) > \delta|) \leqslant 2\exp\left(-\frac{N_{h,i}(\eta,n)\delta^2}{4\nu_{h,i}^2(n)}\right), \tag{32}$$

where $\nu_{h,i}^2(n) = \sigma^4 + 2\sigma^2\zeta_{h,i}(n)$ is the variance of $y(n_{h,i})$, and $\zeta_{h,i}(n) = |\mathbf{h}(n)\mathbf{f}_{h,i}|^2$.

Proof: Our proof follows the steps from [46, Appendix E] and [47, Section 2.1.3]. Note that $\mu_{h,i} = \zeta_{h,i} + \sigma^2$, with the moment generating function of $y(n_{h,i})$ and dropping the time dependence temporarily, we write,

$$\mathbb{E}(\exp(\lambda(y(n_{h,i}) - \mu_{h,i}))) = \frac{\exp(-\lambda\mu_{h,i})}{1-\sigma^2\lambda}\exp\left(\frac{\lambda\zeta_{h,i}}{1-\sigma^2\lambda}\right) \tag{33}$$

$$\leqslant \exp(\sigma^4\lambda^2)\exp(2\zeta_{h,i}\sigma^2\lambda^2) \tag{34}$$

$$= \exp\left(\frac{2\nu_{h,i}^2\lambda^2}{2}\right), \tag{35}$$

where (34) holds when $|\lambda| < 1/(2\sigma^2)$. We use (35) with the Cramer-Chernoff method to derive our concentration bound. The full steps are available in Appendix **??** of the supplementary material. In particular, we are interested in the empirical mean of $y(n_{h,i})$ over time (15), hence,

$$\mathbb{P}(\hat{\mu}_{h,i}(\eta,n) - \mu_{h,i}(n) \geqslant \delta) \leqslant \exp\left(-\frac{N_{h,i}(\eta,n)\delta^2}{4\nu_{h,i}^2(n)}\right). \tag{36}$$

A union bound completes our proof. ∎

Lemma 1 enables us to write the concentration expression using our choice of confidence term (17) and exploration rate (19) for the next lemma.

Lemma 2. *Let $\{y(n_{h,i}) : n_{h,i} \geqslant 2\}$ be the sequence of independent and identically distributed (i.i.d.) random variables in Lemma 1, then for any $B, C \geqslant 1$, $0 < \delta \leqslant \nu_{h,i}^2(n)/\sigma^2$, exploration rate $\beta(\eta,n,\delta)$ in (19), and confidence term $D_{h,i}(\eta,n)$ in (17),*

$$\mathbb{P}(|\hat{\mu}_{h,i}(\eta,n) - \mu_{h,i}(n)| \geqslant D_{h,i}(\eta,n)) \leqslant 3\exp(-\beta(\eta,n,\delta)). \tag{37}$$

Proof: We use the one-sided version of (32) from our result in Lemma 1 with a constant $B \geqslant 1$,

$$\mathbb{P}\left(\hat{\mu}_{h,i}(\eta,n) - \mu_{h,i}(n) \geqslant \sqrt{\frac{4B\nu_{h,i}^2(\eta,n)\beta(\eta,n,\delta)}{N_{h,i}(\eta,n)}}\right) \leqslant \exp(-\beta(\eta,n,\delta)), \tag{38}$$

and the result in [48, Theorem 10] with $C \geqslant 1$ to bound the difference between standard deviation $\nu_{h,i}$ and its empirical estimate, $\hat{\nu}_{h,i}(\eta,n)$, as

$$\mathbb{P}\left(\nu_{h,i}(n) > \hat{\nu}_{h,i}(\eta,n) + \sqrt{\frac{2C\beta(\eta,n,\delta)}{N_{h,i}(\eta,n)-1}}\right) \leqslant \exp(-\beta(\eta,n,\delta)). \tag{39}$$

By replacing $\nu_{h,i}^2(n)$ in (38) with

$$\hat{\nu}_{h,i}(\eta,n) + \sqrt{\frac{2C\beta(\eta,n,\delta)}{N_{h,i}(\eta,n)-1}}, \tag{40}$$

simplifying, and using union bounds, we obtain our result. The full steps are available in Appendix **??** of the supplementary material. ∎

The concentration expressions in (32) and (37) do not explicitly account for the changing mean reward, $\mu_{h,i}(n)$, over time. However, our choice of confidence term (17) incorporates the empirical variance, which past works have shown can suffice to adjust for the dynamic rewards [48], [49].

B. Confidence with Alignment Time

Determining the likelihood of event $\mathcal{A}_h$ (31) requires a probabilistic description of the angle, $\theta(n)$, over time. With the random variable model and distribution in hand (full derivation in Appendix **??** of the supplementary material), we determine the likelihood of $\theta(n)$ remaining in the region, $\mathcal{R}_{h,i} = [\bar{\phi}_{h,i} - \phi_{\text{bw},h}/2, \bar{\phi}_{h,i} + \phi_{\text{bw},h}/2]$, under the kinematic motion described in Section II-B. We express the probability of alignment after n timesteps as

$$\mathbb{P}\left(\left|\bar{\phi}_{h,i} - \theta(n)\right| \leqslant \frac{\phi_{\text{bw},h}}{2}\right) = \frac{\sqrt{2}\sigma_n}{\phi_{\text{bw},h}\sqrt{\pi}}\left(\exp\left(-\frac{\phi_{\text{bw},h}^2}{2\sigma_n^2}\right) - 1\right) + \operatorname{erf}\left(\frac{\phi_{\text{bw},h}}{\sqrt{2}\sigma_n}\right), \quad (41)$$

with

$$\sigma_n^2 \triangleq \frac{\tau^4}{4}\left(\frac{4n^3}{3} - 4n^2 + \frac{11n}{3} - 1\right)\sigma_u^2 + \tau^2(n-1)\sigma_u^2, \quad (42)$$

where τ is the time difference, in seconds, between $n-1$ and n.[9] We use (41) with the bounds on complexity of DBZ, which we determine in Section V-E, to characterize the limits of kinematic motion that DBZ is capable of performing. We must choose sample window lengths, $\boldsymbol{\eta}$, at each level, h, such that

$$[\boldsymbol{\eta}]_h < N_h^{\text{a}}. \quad (43)$$

Offline numerical methods provide a means to select elements of $\boldsymbol{\eta}$ that meet the criteria of (43). Our following lemma establishes guarantees on correctness when we choose $[\boldsymbol{\eta}]_h$ properly. In the following lemma, we combine Lemma 2 with our new insights on event $\mathcal{A}_h$ to show confidence of correct beamforming vector selection with a ME.

Lemma 3. *With the choice of $[\boldsymbol{\eta}]_h < N_h^{\text{a}}$ such that $\mathbb{P}(\mathcal{A}_h) \leqslant \delta/(2H)$, under Assumptions 1 and 2, $\mathcal{B}_h$ and $\mathcal{A}_h$ for all $1 \leqslant h \leqslant H$ hold with probability $1-\delta$.*

Our proof shows that $P(\mathcal{B}_h \cap \mathcal{A}_h, \forall 1 \leqslant h \leqslant H) > 1-\delta$ over all time steps, n, and all arms, i.

[9]We use the error function defined as $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z \exp(-t^2)dt$.

Proof: We apply (37) from Lemma 2 for one level, h, where each (h,i) has $N_{h,i}(\eta,n) = u \geqslant 2$ samples taken. Then, using a union bound over all levels,

$$\mathbb{P}(\mathcal{B}_1^c \cup \cdots \cup \mathcal{B}_H^c) \leqslant \mathbb{P}(\mathcal{B}_1^c) + \cdots + \mathbb{P}(\mathcal{B}_H^c) \quad (44)$$

$$\leqslant \sum_{n=1}^{\infty}\sum_{h=1}^{H}\sum_{i:(h,i)\in\mathcal{I}_h}\sum_{u=1}^{n} 3\exp(-\beta(\eta,n,\delta)) \quad (45)$$

$$\leqslant \sum_{h=1}^{H}\frac{|\mathcal{I}_h|\,\delta}{2\mathcal{I}^H} = \frac{\delta}{2}. \quad (46)$$

With the appropriate choice of η, we combine (46) with

$$\mathbb{P}(\mathcal{A}_1^c \cup \cdots \cup \mathcal{A}_H^c) \leqslant \mathbb{P}(\mathcal{A}_1^c) + \cdots + \mathbb{P}(\mathcal{A}_H^c) \quad (47)$$

$$\leqslant \sum_{h=1}^{H}\frac{\delta}{2H} = \frac{\delta}{2}, \quad (48)$$

from which we conclude that $P(\mathcal{B}_h \cap \mathcal{A}_h, \forall 1 \leqslant h \leqslant H) > 1-\delta$. ∎

C. Sampling Strategy Performance

DBZ adapts the sampling and termination policy of [41] in order to zoom in. We adapt [41, Lemma 4, Lemma 2 and Corollary 1] to show that at each level h, $(h,u(n))$ and $(h,\gamma(n))$ are good choices for sampling, where the policy is *greedy* toward the termination criteria (26). DBZ differs from [41] in the confidence term (17) and exploration rate (19), which include the empirical variance, $\nu_{h,i}^2(\eta,n)$ (18), and the total number of arms, $\mathcal{I}^H$.[10] The operation of DBZ consists in playing *independent* MAB games at each level h dictated by the pruning vector $\mathbf{p}$, hence each lemma extends to all choices of $\mathbf{p}$.

Lemma 4. *Let $S(n) \in \{u(n), \gamma(n)\}$ denote the arm pulled at time step n. At each time step $n \geqslant 2$,*

$$S(n) = u(n) \implies L_{h,u(n)}(\eta,n) \leqslant L_{h,\gamma(n)}(\eta,n), \quad (49)$$

$$S(n) = \gamma(n) \implies U_{h,u(n)}(\eta,n) \leqslant U_{h,\gamma(n)}(\eta,n), \quad (50)$$

and if $S(n) = (h,i)$ then

$$G_{h,\gamma(n)}(n) \leqslant 2D_{h,i}(\eta,n-1). \quad (51)$$

Proof: The proof requires basic handling of each case, as outlined in [41], applied to a single level h. We provide the detailed proof in Appendix **??** of the supplementary material. ∎

From Lemma 4, we provide an upper bound on $G_{h,\gamma(n)}(\eta,n)$ adapted from [41, Lemma 2]. The upper bound allows us to derive an expression in the Section V-E to describe the complexity of the DBZ algorithm.

Lemma 5. *On event $\mathcal{B}_h$, if $(h,i) \in \{(h,u(n)),(h,\gamma(n))\}$ at time step $n \geqslant 2$, then*

$$G_{h,\gamma(n)}(\eta,n) \leqslant \min\{0, 2D_{h,i}(\eta,n-1) - \Delta_{h,i}(n)\} + 2D_{h,i}(\eta,n-1). \quad (52)$$

[10]$\mathcal{I}^H = \sum_h |\mathcal{I}_h|$ is the total quantity of beamforming vectors participating in MAB games at all levels, and is fixed for any codebook.

Proof: Similar to Lemma 4, this proof requires bookkeeping to analyze each statement, as outlined in [41], at level h. We provide the detailed steps in Appendix **??** of the supplementary material. ∎

D. Zooming In and Zooming Out

Because of the ME motion (5), the unique maximum mean reward, $\mu_h^*(n)$, and paths within the reward structure change over time (see Assumptions 1 and 2). Section IV-B describes the mechanics uses to adapt the beamwidth to compensate for ME motion, but we must ensure correct decisions to zoom in or out. The following lemma, adapted from [41, Lemma 1] ensures an arm, (h,i), will not be mistakenly chosen as an ϵ_h-optimal arm and zoomed in on, under event $\mathcal{B}_h$.

Lemma 6. *If $\mathcal{B}_h$ holds, for any $(h,i) \notin \{(h,i) : \mu_{h,i}(n) + \epsilon_h \geqslant \mu_h^*(n)\}$, $G_{h,i}(\eta,n) \geqslant \epsilon_h$ for all $n \geqslant 2$.*

Proof:

$$G_{h,i}(\eta,n) = \max_{j\neq i} U_{h,j}(\eta,n-1) - L_{h,i}(\eta,n-1) \tag{53}$$
$$\geqslant \max_{j\neq i} \mu_{h,j}(n) - \mu_{h,i}(n) \tag{54}$$
$$= \mu_h^*(n) - \mu_{h,i}(n) > \epsilon_h, \tag{55}$$

where (54) is from $\mathcal{B}_h$. (55) comes from the fact that $i \neq i^*$ and the definition of an ϵ_h-optimal arm. ∎

Say DBZ terminates with (h',i') from a previous level $h' < h$, and produces threshold $[\boldsymbol{\kappa}]_{h'+1} = L_{h',i'}(\eta,n-1) + A\epsilon_{h'}$. Let n and n' represent time steps at levels h and h', respectively. Complementing Lemma 6, our next lemma ensures that DBZ zooms out according to changes in reward structure due to motion.

Lemma 7. *If $\mathcal{B}_h$ and $\mathcal{B}_{h'}$ hold, with threshold $[\boldsymbol{\kappa}]_{h'+1}$ in (27) based on termination at level h' with (h',i'), and $A \leqslant 1$, DBZ zooms out correctly with probability greater than $1-\delta$ if $\nexists (h,i) \in \mathcal{I}_h$ such that $\mu_{h,i}(n) > \mu_{h',i'}(n') + \epsilon_{h'}$.*

Proof: If DBZ zooms out from level h, we have that $U_{h,\gamma(n)}(\eta,n-1) < [\boldsymbol{\kappa}]_{h'+1}$, and therefore

$$\mu_{h,i}(n) \leqslant \max_{i:(h,i)\in\mathcal{I}_h} \mu_{h,i}(n) \tag{56}$$
$$\leqslant U_{h,\gamma(n)}(\eta,n-1) \tag{57}$$
$$< [\boldsymbol{\kappa}]_{h'+1} \tag{58}$$
$$= L_{h',i'}(\eta,n'-1) + A\epsilon_{h'} \tag{59}$$
$$\leqslant \mu_{h',I}(n') + \epsilon_{h'}. \tag{60}$$

The relationship of (56) to (57) and (59) to (60) come from event $\mathcal{B}_h$ and $\mathcal{B}_{h'}$, respectively, where Lemma 2 shows both events hold with probability greater than $1-\delta$. ∎

Together, Lemmas 6 and 7 show that despite the time-varying mean rewards, DBZ will correctly zoom in and out with at least probability $1-\delta$ under event $\mathcal{B}_h$ for all h.

E. Sample Complexity

We now provide an analysis of the sample complexity of DBZ to zoom in at each level. The sample complexity enables setting sample window lengths $\boldsymbol{\eta}$ large enough to accommodate the number of samples to zoom in (26). For zooming in, DBZ exploits the structure induced by the hierarchical codebook by reducing the overall total number of arms considered along a path (see Assumption 2), $\mathcal{I}^H$, by order of the logarithm of the number of beamforming vectors required in traditional MAB strategies using the narrowest beams [15], [26]. The reduction in beamforming vectors considered directly attributes to an overall reduction in sample complexity. However, the variance and individual spacing between mean rewards of arms also play a significant role in determining overall complexity. Let $\Delta_{h,i,\epsilon}(n) \triangleq \max\{(\Delta_{h,i}(n)+\epsilon_h)/4, \epsilon_h/2\}$, and

$$\aleph_{h,\epsilon}(n) \triangleq \sum_{i:(h,i)\in\mathcal{I}_h} \frac{2B\nu_{h,i}^2(n) + 2\sqrt{2BC}\Delta_{h,i,\epsilon}(n)}{\Delta_{h,i,\epsilon}^2(n)} + \frac{\sqrt{4B^2\nu_{h,i}^4(n) + 2\sqrt{2C}B^{3/2}\nu_{h,i}^2(n)\Delta_{h,i,\epsilon}(n)}}{\Delta_{h,i,\epsilon}^2(n)}. \tag{61}$$

The relevance of $\aleph_{h,\epsilon}(n)$ is justified by the following lemma.

Lemma 8. *If $\mathcal{B}_h$ holds, DBZ ensures that the number of samples of beamforming vector (h,i) after n total samples at level h, satisfies*

$$N_{h,i}(\eta,n) \leqslant \frac{2B\nu_{h,i}^2(n) + 2\sqrt{2BC}\Delta_{h,i,\epsilon}(n)}{\Delta_{h,i,\epsilon}^2(n)}\beta(\eta,n-1,\delta) + \frac{\sqrt{4B^2\nu_{h,i}^4(n) + 2\sqrt{2C}B^{3/2}\nu_{h,i}^2(n)\Delta_{h,i,\epsilon}(n)}}{\Delta_{h,i,\epsilon}^2(n)} \times \beta(\eta,n-1,\delta) + 2, \tag{62}$$

or rounded to the next largest integer, $N_{h,i}^*(\eta,n) = \lceil N_{h,i}(\eta,n)\rceil$.

Proof: The proof involves writing (52) and replacing with our expression for $D_{h,i}(\eta,n-1)$ in (17), then solving for $N_{h,i}(\eta,n)$. The full steps are available in Appendix **??** of the supplementary material. ∎

F. Configuring DBZ

This section provides DBZ users with a practical methodology for selecting a sample window length, $\boldsymbol{\eta}$, pruning vector, $\mathbf{p}$, and parameter ϵ. To set $\boldsymbol{\eta}$ and $\mathbf{p}$, we use Lemma 8 that describes the total number of samples required at each level, $N_{h,i}^*(\eta,n)$, which scales directly with noise variance, σ_v^2. In order for DBZ to take sufficient samples such that it meets either the criteria of (26) or (28), we require elements of $\boldsymbol{\eta}$ large enough, such that for a single element η,

$$\eta \geqslant \sum_{i:(h,i)\in\mathcal{I}_h} N_{h,i}^*(\eta,n). \tag{63}$$

Note that $\sum_{i:(h,i)\in\mathcal{I}_h} N_{h,i}(\eta,n) = \eta$ if $n \geqslant \eta$ and n otherwise. We obtain an estimate of how to set η by further

analyzing (62), where the total number of samples required at each level is

$$\eta = \sum_{i:(h,i)\in\mathcal{I}_h} N_{h,i}(\eta, n) \leqslant \aleph_{h,\epsilon}(n) \log\left(\frac{15 N_H \eta^4}{2\delta}\right) + 2\left|\mathcal{I}_h\right| \tag{64}$$

and has the closed form solution to suggest the value,

$$\eta_{\text{est}} = \left\lceil -4\aleph_{h,\epsilon}(n) W\left(-\frac{\exp\left(-\frac{2|\mathcal{I}_h|-1}{4\aleph_{h,\epsilon}(n)}\right)}{4\aleph_{h,\epsilon}(n)\left(\frac{15\mathcal{I}^H}{2\delta}\right)^{1/4}}\right)\right\rceil + 1, \tag{65}$$

where $W(\cdot)$ is the Lambert-W function.[11] The sample window length should be chosen such that

$$[\boldsymbol{\eta}]_h \geqslant \eta_{\text{est}} \tag{66}$$

In cases of extreme motion with very large σ_u and/or especially low SNR with large σ_v, we conclude that DBZ delivers poor performance. When $\eta_{\text{est}} > N_h^{\text{a}}$, DBZ cannot guarantee selection of ϵ-optimal beamforming vectors with at least probability $1-\delta$. For practical implementation, a user should choose σ_u in (42) such that it approximates the highest angular acceleration possible by the ME intended to track.

We perform optimization of the pruning vector, $\mathbf{p}$, in an offline manner to optimize utilization of the beamforming codebook. We show in Section V that the choices of $\mathbf{p}$ generalize over a broad range of SNR. Minimizing (65) over the range of possible path angles provides an assessment of which pruning vector, $\mathbf{p}$, is optimal. We require the expected number of samples at level h, $\mathbb{E}_{\theta_1}(\eta_{\text{est}}(\theta))$. "Averaging" over the range of angles Φ eliminates dependence on the angle. Furthermore, the sparsity of the mmWave channel allows us to focus on the dominant path, θ_1 [43]. The vector $\mathbf{p}^*$ minimizes the average complexity, such that

$$\mathbf{p}^* = \underset{\mathbf{p}}{\operatorname{argmin}}\, \mathbb{E}_{\theta_1}\left(\sum_{h:[\mathbf{p}]_h=1} \eta_{\text{est,h}}(\theta)\right), \tag{67}$$

and we estimate the expected number of samples,

$$\mathbb{E}_{\theta_1}\left(\sum_{h:[\mathbf{p}]_h=1} \eta_{\text{est,h}}(\theta)\right), \tag{68}$$

numerically. Our numerical simulations in the next section show the samples required for initial alignment with different choices of $\mathbf{p}$ for comparison. We include example code for computing $\boldsymbol{\eta}$ and $\mathbf{p}$ in our source code [50].

DBZ uses the parameter ϵ to compensate for cases with especially small $\Delta_{h,i}(n)$, when two mean rewards are very close in value. The case of small $\Delta_{h,i}(n)$ occurs when $\theta_1 \approx \bar{\phi}_i \pm \phi_{\text{bw},h}/2$ or Non-LOS (NLOS) scenarios where there is no clear dominant path, causing the RSRP (4) of multiple beamforming vectors to be very similar. As a reminder, the ϵ parameter in the termination criteria allows DBZ to terminate with a sub-optimal arm (H,i), such that $\mu_H^* \leqslant \mu_{H,i} + \epsilon$. The sub-optimal choice impacts the relative spectral efficiency with respect to ϵ as

$$\tilde{\xi}_{h,i} \triangleq \frac{\log_2\left(1 + (\zeta_{h,i^*}(n) - \epsilon)/\sigma_v^2\right)}{\log_2\left(1 + \zeta_{h,i^*}(n)/\sigma_v^2\right)} \tag{69}$$

for $h = H$ and for all $\epsilon > 0$. We set ϵ such that $\tilde{\xi}_{h,i} > .95$ for all h. In practice, $\zeta_{h,i^*}(n)$ corresponds to some maximum RSRP, while ϵ denotes the penalty allowed with communication persisting. We note that choosing an ϵ-optimal arm is unique to DBZ compared to existing algorithms [14], [51] that fall victim to high complexity with small $\Delta_{h,i}(n)$. With ϵ_h, we use our scaling of ϵ with respect to the gain at level h, $g^{-(H-h)}$, for each subsequent level of the hierarchical beamforming codebook $\mathcal{F}^H$. We expect $\Delta_{h,i}(n)$ to be smaller at lower levels, or overall in NLOS scenarios. By scaling ϵ to ϵ_h for the corresponding level, h, we ensure that there is no unnecessarily high penalty to relative spectral efficiency incurred for our beamforming vector selection at termination.

VI. Numerical Simulations

Our numerical simulations assess the ISAC performance of DBZ to quickly align, i.e., choose a beamforming vector at level H, and adjust the beamforming pattern width over time to compensate for motion while communicating. Our simulation source code is available at [50].

A. Methodology for Initial Alignment Simulations

We execute each simulation by first making K uniformly random selections $\theta_k \in \Phi$, each representing the k^{th} path. We use a unique random number generator seed for each individual simulation that we denote with index ℓ. The K-length vector of angles chosen for simulation ℓ is denoted $\boldsymbol{\theta}_\ell$ with a corresponding vertex (H, i_ℓ^*). We use $\boldsymbol{\theta}_\ell$ to then construct the array response (2). We take samples by applying beamforming vectors to the channel model observations, as in (4), that are chosen based on the algorithm policy. Each simulation terminates after the stopping criteria (26) is met. We compare the performance of DBZ across several SNR values with various pruning vectors, $\mathbf{p}$ (which we denote by their decimal values), and directly with HPM from [13] and 2PHTS from [14].[12] The HPM algorithm acts a baseline of performance in utilizing perfect channel knowledge in the posterior computations to deploy the hierarchical codebook. Another potential comparison candidate algorithm, HBA, aggressively searches the range of Φ, sacrificing performance under lower-SNR conditions to terminate quickly. 2PHTS adapts the state-of-the-art TAS MAB framework using an approximation of the stochastic channel model that works for high SNR. We dynamically determine the number of total simulations required, L, by utilizing the Wilson score [52] interval width. Further details of the confidence intervals are

[11] The Lambert-W function enables the relation $x_h\eta \geqslant \log(y_h\eta) \iff \eta \leqslant -\frac{1}{x_h} W\left(-\frac{x_h}{y_h}\right)$, where $x_h = 1/(4\aleph_{h,\epsilon}(n))$ and $y_h = (15\mathcal{I}^H/(2\delta))^{1/4} \exp\left((2|\mathcal{I}_h| - 1)/(4\aleph_{h,\epsilon}(n))\right)$.

[12] Note that there is some degradation at high SNR for HPM [13] due to not perfectly compensating for the multi-path effects. Additionally, we could only simulate the behavior of 2PHTS in the high-SNR regime because of numerical issues intrinsic to the algorithm.

TABLE II: Details on pruning vector values, $\mathbf{p}$.

p_{dec}	$\mathbf{p}$	h **Traversed**
0	0000001	7
3	0000111	5, 6, 7
4	0001001	4, 7
7	0001111	4, 5, 6, 7
8	0010001	3, 7
63	1111111	All h

available in Appendix **??** of the supplementary material. Let $T_h(\ell)$ denote the samples required for level h in simulation ℓ, the average sample complexity, or number of beamforming vectors required, is

$$\hat{T}(L) = \frac{1}{L}\sum_{\ell=1}^{L}\sum_{h:[\mathbf{p}]_h=1} T_h(\ell). \tag{70}$$

For the initial alignment performance, algorithms utilize a common beamforming codebook with $H = 7$ levels (128 pointing angles at the finest resolution) organized by a binary tree graph with $M = 128$ antenna elements in a ULA. We design the beamforming architecture to support as few as a single RF chain in a HAD configuration based on the design in [3]. The gain parameter is set as $g = 10^{.2}$ which corresponds to 2 dB of gain per level with the increasingly narrow beams. We fix $P_k = 1$ and assume no knowledge of the channel SNR. We also do not use any knowledge of the channel fading factors, $\alpha_k(n)$ (1), in DBZ. Our results show that DBZ is robust to the time-varying $\alpha_k(n)$. The sequence $\mathbf{p}$ is chosen as decimal values $p_{\text{dec}} \in \{0, 3, 4, 7, 8\}$, identified with the methodology in Section V-F to be a good set of $\mathbf{p}$ to compare. We summarize the details of each selection of $\mathbf{p}$ in Table II. We use the convention "<algorithm>p_{dec}", i.e., DBZ7, to indicate the algorithm and selection of pruning vector.

B. Results and Discussion

Our initial alignment experiments investigate the overall complexity (70), which is the key metric for the fixed confidence best arm identification setting. We emphasize that the ME is NOT mobile during these initial alignment simulations, as to have a fair comparison with other algorithms. We also verify that the expected relative spectral efficiency after n samples and the algorithm chooses a beamforming vector,

$$\xi(n, L) \triangleq \frac{1}{L}\sum_{\ell=1}^{L}\frac{\log_2\left(1+\zeta_{h,i^c}(n)/\sigma_v^2\right)}{\log_2\left(1+\zeta_{H,i^*}(n)/\sigma_v^2\right)}, \tag{71}$$

is obtained after obtaining samples with chosen ϵ. Fig. 4 and 5 provide a comparison of the sample complexity and resulting relative spectral efficiency for several mmWave beam alignment algorithms:

- DBZ with several configurations of the pruning vector, $\mathbf{p}$, along with two values of ϵ.
- HPM from [13] utilizing perfect CSI of both the channel fading coefficient, $\alpha_1(n)$, and noise variance, σ_v^2.
- HBA from [15], which bisects the search space according to the MAB policy in [29].
- Hierarchical Optimal Sampling of Unimodal Bandits (HOSUB) from [53] that has operates as a fixed-budget (or fixed number of samples) algorithm using the MAB framework in [27] to explore the hierarchical codebook graph. We show the performance with two different budget constraints, 50 and 100.

As anticipated, HPM provides a baseline for performance in that it optimally exploits the induced structure by using CSI to compute the posteriors at each time step. In general, we anticipate many of the algorithms that compute the explicit distributions [36], [51] offer similar performance, but with the price of significant computational overhead to compute the posteriors. At very low SNR, an exhaustive search (DBZ0) outperforms any other DBZ variation in Fig. 4. This is expected, in fact, works such as [14] hinge on the assumption of exclusively operating in a high-SNR regime. Our results show that values of SNR roughly between -6 to 6 are the target SNR regimes in which DBZ achieves better complexity than an exhaustive search. Fig. 5 shows a significant reduction in relative spectral efficiency at low SNR for HBA and HOSUB, which both sacrifice some performance for lower complexity, shown in Fig 4. DBZ lowers its complexity by utilizing larger values of ϵ, however, there is a corresponding loss in relative spectral efficiency shown in Fig. 5.

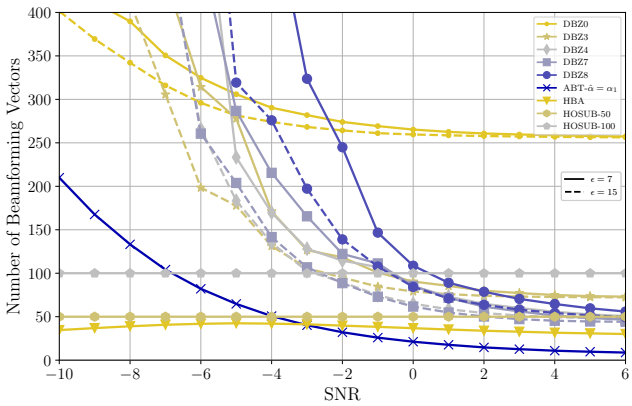

Fig. 4: Comparison of complexity at various SNR.

We provide one last numerical result for initial alignment at SNR $=$ 20 dB (a high SNR regime) to compare the performance of DBZ to that of TAS methods in Table III. We adapt the TAS framework in [54] for identifying an ϵ-optimal arm, and similar to [14], apply TAS over subsequent levels, h, as shown in Table II. We also use the assumption in [14], in which the observation is close to a Heteroscedastic Gaussian to compute the relative entropy in the TAS steps, and apply our scaling of ϵ as ϵ_h at each level. As expected, HPM achieves the best results given the full CSI. A particular point of interest is that strategies considering fewer arms, $\mathcal{I}^H$, perform significantly better at high SNR. One concludes that at high SNR, $\mathbf{p}$ should be chosen to minimize $\mathcal{I}^H$. While TAS methods perform better overall, the number of samples for DBZ63 and DBZ31 are only marginally worse than TAS63 and TAS31. Some of the algorithms in Table I do not have an

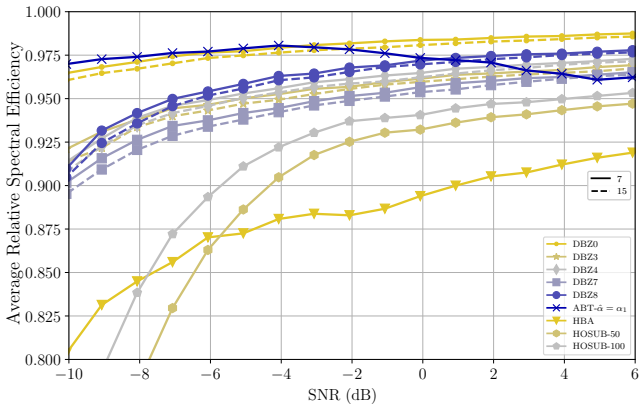

Fig. 5: Comparison of resulting relative spectral efficiency at various SNR.

TABLE III: Complexity in high SNR regime.

DBZ4	DBZ7	DBZ31	DBZ64	TAS4	TAS7	TAS31	TAS64	HPM
57.0	53.7	30.4	30.8	26.0	26.0	20.0	21.0	8.2

explicit initial alignment component or guarantees on accuracy in the algorithm, [9], [10], [19]. For the tracking simulations in the next subsection, we assume [9], [10], [19] incur an $O(I)$ sample complexity with an exhaustive search of all narrow beams to initially align. There may be space in future work to combine initial alignment approaches like DBZ, HPM, or 2PHTS with [9], [19] to enhance algorithm performance.

C. Methodology for Tracking Simulations

We provide a series of numerical simulations to demonstrate the performance of DBZ under different channel SNR, σ_v, and magnitude of motion, σ_u (and more extensively in Appendix **??** of the supplementary material). We fix the interval in which samples are taken, $\tau = 1$, and execute each simulation by first choosing $\theta_k(1)$ as in our initial alignment simulations, according to a random number generator seed, ℓ. We take a single sample (3) at each time step by applying beamforming vectors to the channel model observations, as in (4), that are chosen based on the flowchart in Fig. 3. After each sample the ME undergoes the kinematic motion transition in (5). We execute the main algorithm loop for DBZ (After input and parameter initialization in Fig. 3) until a specified number of time steps occur, N. We perform L simulations of N time steps, and calculate the average relative spectral efficiency at each time step, n, (71). The indices (h, i^c) in the numerator of (71) corresponds to the beam currently being used for communication. DBZ adapts to the changing $\theta(n)$ by broadening (zooming out) and narrowing (zooming in) the beam used to communicate on the events in lines 21 and 5, respectively. We use a ternary hierarchical codebook with $\mathcal{I}_1 = 5$, with depth $H = 4$, and each beam splits into 3 narrower beams, creating 135 narrow beams at $h = H$. We use $\mathbf{p} = [1, 1, 1, 1]$ for all tracking simulations. The degradation of each algorithm's performance at later time steps comes from the ME possibly accelerating to reach faster speeds (5), making that tracking task more difficult over time.

D. Comparison of Algorithms

We provide a performance comparison across several algorithms for mmWave beam tracking by assessing the time-average relative spectral efficiency, of (72),

$$\frac{1}{N}\sum_{n=1}^{N}\xi(n, L). \tag{72}$$

In particular, we use the following algorithms for comparison in simulation:

- The Active Beam Tracking (ABT) algorithm from [20], [22], which acts as our baseline algorithm by exploiting full CSI and knowledge of the ME motion to compute the Bayesian posteriors and select beamforming vectors. We assess two variations, one in which the fading coefficient, $\alpha_1(n)$, is known and one variation that uses a noisy estimate of $\alpha_1(n)$.
- PF approach from [19], which uses the covariance of the particles to broaden or narrow the beam by activating a specified number of antenna elements. To offer a more fair comparison, we assess the effective beamwidth produced by the number of elements, and use a level in our ternary codebook $\mathcal{F}^H$ that most closely matches the effective beamwidth.
- MAB approach in [10], which periodically sweeps neighboring "offset" narrow beams in a different type of MAB application.
- EKF approach in [9], where we use the angle estimations to select the narrow beamforming vectors.

Our implementation of each algorithm is in the source code [50]. Each algorithm has different trade-offs with respect to the characteristics listed in Table I. Fig. 6 and 7 show the performance of each algorithm in LOS and NLOS scenarios with different severity of motion, σ_u. In our LOS scenario, the dominant path is 10 dB above the others, where the NLOS has no clear dominant path. We see that DBZ outperforms all other algorithms except ABT [20], as expected. The adaptive beamwidth control for the PF approach, [19], allows for better performance than the offset sweep in [10] or the Kalman Filter (KF) in [9]. However, the adaptive beamwidth control for DBZ exceeds that of the PF. Combination of the PF or KF with DBZ could yield a potent algorithm for mmWave tracking. In the next section, we take a closer look at the results of comparing DBZ with the Bayesian algorithm, ABT from [20]. We show there are instances where DBZ indeed performs better if ABT does not have access to exquisite channel information, failing to be CSI adaptive.

E. Comparison to Bayesian Method

Our first experiment compares DBZ performance with an extension of HPM to compensate for motion, ABT [20], [22]. We use ABT as a baseline of performance, the Bayesian framework leverages full channel information to compute the posteriors on beamforming vectors $\mathbf{f}_{H,i}$ after each observation.

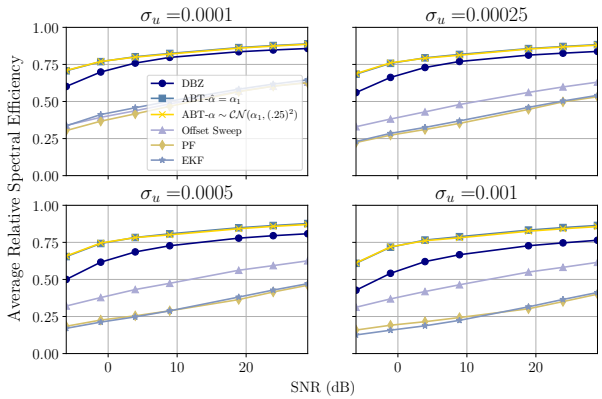

Fig. 6: Comparison of performance between algorithms in LOS scenario.

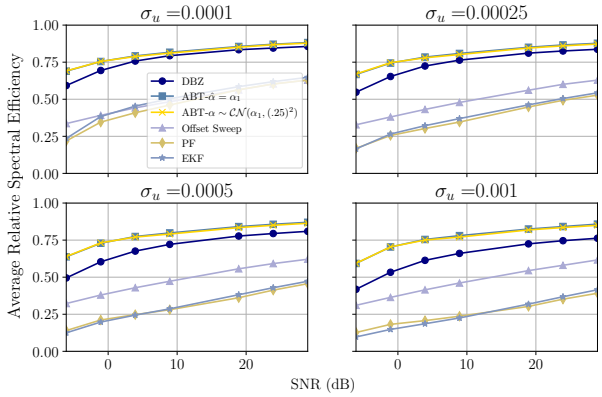

Fig. 7: Comparison of performance between algorithms in NLOS scenario.

The broader beams posteriors are the sum of the posteriors for narrower ones. The framework in [22] shows a way to optimize RS (pilot signals) or data sent based on optimizing spectral efficiency. However, for this comparison, we fix the interval in which RS are sent. We calculate the posterior for the measurement (3), which is corrupted by Additive White Gaussian Noise (AWGN) under multiplicative fading coefficient $\alpha_1(n)$. We also apply the density for the entity motion, (see (**??**) in Appendix **??** of the supplementary material), to the posterior. Fig. 8 shows our results for DBZ using sample windows set by the estimated complexity, η_{est}, with a subset of SNRs compared with the performance of ABT. ABT performs extremely well when the fading coefficient, $\alpha_1(n)$, is used in the computation of the posterior. In practice, this comes from a method to make a precise estimate of the coefficient. To assess performance when the estimation is in slight error, we choose the fading coefficient as a random variable distributed as $\mathcal{CN}(\alpha_1(n), (.5)^2)$. We see a slight degradation in performance with the error in estimation of the fading coefficient. What may be more interesting however, is the performance disparity between high and low SNR (SNR$= 14$ versus SNR$= -6$), in that one would expect better performance at higher SNR, but the opposite is shown in Fig. 8. This is due to the imperfect posterior computed at high SNR creating "overconfidence" in the selection of narrower beamforming vectors. At lower SNR, ABT is more discerning (broader distributions) in its choices to narrow or broaden the beam, hence there is less emphasis on accurate estimations of $\alpha_1(n)$. We see DBZ is competitive with ABT given no channel knowledge other than the SNR to compute the sampling window lengths, $\boldsymbol{\eta}$. DBZ only requires $O(|\mathcal{I}_h| = 3)$ Floating-Point Operation (FLOP)s ($O(|\mathcal{I}_1| = 5)$ in our case) versus ABT with $O(128)$ FLOPs (with the codebook used), for each of the algorithm's computational cost at each sample. The $O(128)$ in ABT comes from the need to update each posterior for each beamforming vector at each sampling iteration with the binary codebook used therein.[13]

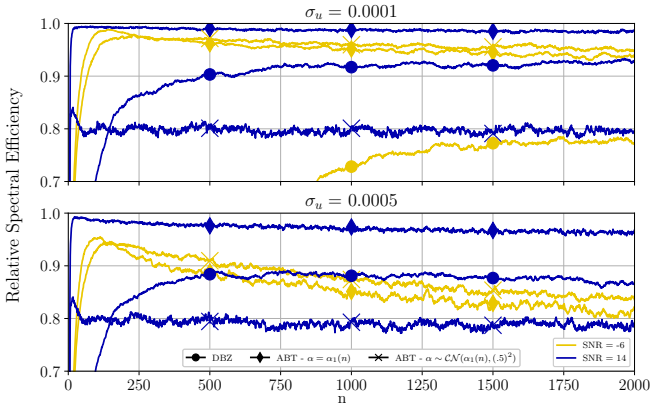

Fig. 8: Comparing performance between ABT with full channel knowledge and DBZ with η set by estimated complexity.

F. Performance in NYU Sim Model

The question of DBZ performance in the presence of realistic multi-path environments remains. We use NYU Sim [33], [34] to generate $L = 100$ unique spatially consistent trajectories of a moving entity within Φ. In using NYU Sim, we compute the estimated relative spectral efficiency (71) over each trajectory consisting of $N = 600$[14] time steps and create an average result for each scenario and track, i.e. `UMa : Linear`. We provide the full list of parameters to configure NYU Sim in Appendix **??** of the supplementary material. DBZ (and other algorithms alike) struggle to handle drastic large-scale fading and outage models where there may be variations of up to ~ 50 dB of power between each time step. This is especially true in urban cases (UMa and UMi). We assume an analog front end that applies an Automatic Gain Control (AGC) mechanism, which we model here as normalization of the channel vector (1), $\sqrt{M}\mathbf{h}(n)/|\mathbf{h}(n)|$. The urban cases still see swings in receive power that would be indicative that severe multi-path is present, despite normalization. We

[13]The relative spectral efficiency metric normalizes any differences in codebook selection between the two algorithms

[14]This number of timesteps worked out to be an integer number with the actual time, in seconds, between time steps n and the length of the track. See Appendix **??** in the supplementary material.

see DBZ performs relatively well in all scenarios. In particular, the rural scenarios, `RMa`, DBZ matches or exceeds it's performance against the DWNA motion model. The severe multi-path elements in the urban scenarios cause edge cases of the induced structure described in Section III-B, where perturbations induced by noise, even small, cause significant degradation in performance. The consistent spikes and valleys in spectral efficiency at specific time steps come from using the same track, which is especially true in the hexagonal track case.

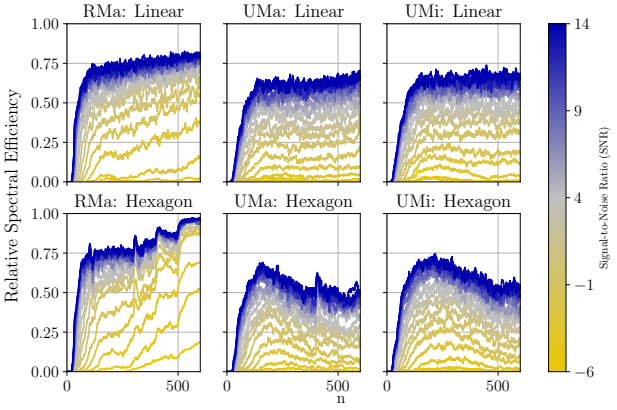

Fig. 9: Comparing performance with NYUsim.

VII. Conclusion

We presented DBZ, an algorithm with low computational overhead, that encompasses all defining features of ISAC. Exploiting the structure induced by the hierarchical codebook, we adapted the MAB best arm identification framework from [41] to handle a ME. Our analysis shows the correctness guarantees on beamforming vector selection. Additionally, we have characterized how to set the sample window lengths based on a DWNA channel model and the complexity expected of the DBZ algorithm. The beamwidth adjustments over time prevent severe outages typically associated with mmWave systems. We show DBZ strikes competitive performance against Bayesian methods exploiting full channel knowledge and ME motion [20]. Finally, our simulations with NYU Sim show DBZ's efficacy in realistic fading environments over several scenarios.

References

[1] Y. Cui, F. Liu, X. Jing, and J. Mu, "Integrating sensing and communications for ubiquitous iot: Applications, trends, and challenges," *IEEE Network*, vol. 35, no. 5, pp. 158–167, 2021.

[2] T. S. Rappaport, G. R. MacCartney, M. K. Samimi, and S. Sun, "Wideband millimeter-wave propagation measurements and channel models for future wireless communication system design," *IEEE transactions on Communications*, vol. 63, no. 9, pp. 3029–3056, 2015.

[3] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 831–846, 2014.

[4] Y. Wang, Z. Wei, and Z. Feng, "Beam training and tracking in mmwave communication: a survey," *arXiv preprint arXiv:2205.10169*, 2022.

[5] A. Soumya, C. Krishna Mohan, and L. R. Cenkeramaddi, "Recent advances in mmwave-radar-based sensing, its applications, and machine learning techniques: A review," *Sensors*, vol. 23, no. 21, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/21/8901

[6] F. Liu, Y. Cui, C. Masouros, J. Xu, T. X. Han, Y. C. Eldar, and S. Buzzi, "Integrated sensing and communications: Towards dual-functional wireless networks for 6g and beyond," *IEEE Journal on Selected Areas in Communications*, pp. 1–42, 2022.

[7] Z. Wei, H. Qu, Y. Wang, X. Yuan, H. Wu, Y. Du, K. Han, N. Zhang, and Z. Feng, "Integrated sensing and communication signals towards 5g-a and 6g: A survey," *IEEE Internet of Things Journal*, 2023.

[8] 3GPP, "TS 38.213: Physical Layer Procedures for Control," 3rd Generation Partnership Project (3GPP), Technical Specification 38.213, 2023, available: 3GPP TS 38.213 Release 17. [Online]. Available: http://www.3gpp.org/DynaReport/38213.htm

[9] V. Va, H. Vikalo, and R. W. Heath, "Beam tracking for mobile millimeter wave communication systems," in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. Washington D.C., USA: IEEE, 2016, pp. 743–747.

[10] J. Zhang, Y. Huang, Y. Zhou, and X. You, "Beam alignment and tracking for millimeter wave communications via bandit learning," *IEEE Transactions on Communications*, vol. 68, no. 9, pp. 5519–5533, 2020.

[11] H.-L. Chiang, K.-C. Chen, W. Rave, M. K. Marandi, and G. Fettweis, "Machine-learning beam tracking and weight optimization for mmwave multi-uav links," *IEEE Transactions on Wireless Communications*, vol. 20, no. 8, pp. 5481–5494, 2021.

[12] D. Zhu, J. Choi, Q. Cheng, W. Xiao, and R. W. Heath, "High-resolution angle tracking for mobile wideband millimeter-wave systems with antenna array calibration," *IEEE Transactions on Wireless Communications*, vol. 17, no. 11, pp. 7173–7189, 2018.

[13] S.-E. Chiu, N. Ronquillo, and T. Javidi, "Active learning and csi acquisition for mmwave initial alignment," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2474–2489, 2019.

[14] Y. Wei, Z. Zhong, and V. Y. Tan, "Fast beam alignment via pure exploration in multi-armed bandits," *IEEE Transactions on Wireless Communications*, pp. 3264–3279, 2022.

[15] W. Wu, N. Cheng, N. Zhang, P. Yang, W. Zhuang, and X. Shen, "Fast mmwave beam alignment via correlated bandit learning," *IEEE Transactions on Wireless Communications*, vol. 18, no. 12, pp. 5894–5908, 2019.

[16] N. Blinn, J. Boerger, and M. Bloch, "mmwave beam steering with hierarchical optimal sampling for unimodal bandits," in *ICC 2021-IEEE International Conference on Communications*. Montreal, Quebec: IEEE, 2021, pp. 1–6.

[17] F. Sohrabi, Z. Chen, and W. Yu, "Deep active learning approach to adaptive beamforming for mmwave initial alignment," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2347 – 2360, 2021.

[18] S. H. Lim, S. Kim, B. Shim, and J. W. Choi, "Deep learning-based beam tracking for millimeter-wave communications under mobility," *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7458–7469, 2021.

[19] H. Chung, J. Kang, H. Kim, Y. M. Park, and S. Kim, "Adaptive beamwidth control for mmwave beam tracking," *IEEE Communications Letters*, vol. 25, no. 1, pp. 137–141, 2020.

[20] N. Ronquillo and T. Javidi, "Active beam tracking under stochastic mobility," in *ICC 2021 - IEEE International Conference on Communications*, June 2021, pp. 1–7.

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

[22] N. Ronquillo, C.-S. Gau, and T. Javidi, "Integrated beam tracking and communication for (sub-) mmwave links with stochastic mobility," *IEEE Journal on Selected Areas in Information Theory*, pp. 94–111, 2023.

[23] D. Tandler, S. Doerner, M. Gauger, and S. ten Brink, "Deep reinforcement learning for mmwave initial beam alignment," in *WSA & SCC 2023; 26th International ITG Workshop on Smart Antennas and 13th Conference on Systems, Communications, and Coding*. VDE, 2023, pp. 1–6.

[24] M. E. Rasekh, Z. Marzi, Y. Zhu, U. Madhow, and H. Zheng, "Noncoherent mmwave path tracking," in *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, ser. HotMobile '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 13–18.

[25] H. Hassanieh, O. Abari, M. Rodriguez, M. Abdelghany, D. Katabi, and P. Indyk, "Fast millimeter wave beam alignment," in *Proceedings*

of the 2018 Conference of the ACM Special Interest Group on Data Communication, ser. SIGCOMM '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 432–445.

[26] M. Hashemi, A. Sabharwal, C. Emre Koksal, and N. B. Shroff, "Efficient beam alignment in millimeter wave systems using contextual bandits," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, Honolulu, HI, USA, 2018, pp. 2393–2401.

[27] R. Combes and A. Proutiere, "Unimodal bandits: Regret lower bounds and optimal algorithms," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 1. Bejing, China: PMLR, 2014, pp. 521–529.

[28] J. Y. Yu and S. Mannor, "Unimodal bandits," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, Bellevue, WA, USA, 2011, pp. 41–48.

[29] S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvári, "X-armed bandits," *Journal of Machine Learning Research*, vol. 12, no. May, pp. 1655–1695, 2011.

[30] A. Garivier and E. Kaufmann, "Optimal best arm identification with fixed confidence," in *29th Annual Conference on Learning Theory*, ser. Proceedings of Machine Learning Research, V. Feldman, A. Rakhlin, and O. Shamir, Eds., vol. 49. New York, New York, USA: PMLR, 23–26 Jun 2016, pp. 998–1027.

[31] E. Kaufmann, O. Cappé, and A. Garivier, "On the complexity of best-arm identification in multi-armed bandit models," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1–42, 2016.

[32] J. Lim, H.-M. Park, and D. Hong, "Beam tracking under highly nonlinear mobile millimeter-wave channel," *IEEE Communications Letters*, vol. 23, no. 3, pp. 450–453, 2019.

[33] S. Ju and T. S. Rappaport, "Simulating motion - incorporating spatial consistency into nyusim channel model," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Chicago, IL, USA, Aug 2018, pp. 1–6.

[34] S. Sun, G. R. MacCartney, and T. S. Rappaport, "A novel millimeter-wave channel simulator and applications for 5g wireless communications," in *2017 IEEE International Conference on Communications (ICC)*, Paris, France, May 2017, pp. 1–7.

[35] 3GPP, "TS 38.211: Physical channels and modulation," 3rd Generation Partnership Project (3GPP), Technical Specification 38.211, 2023, available: 3GPP TS 38.211 Release 17. [Online]. Available: http://www.3gpp.org/DynaReport/38211.htm

[36] S. Noh, M. D. Zoltowski, and D. J. Love, "Multi-resolution codebook and adaptive beamforming sequence design for millimeter wave beam alignment," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5689–5701, 2017.

[37] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave mimo systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 436–453, 2016.

[38] 3GPP, "TS 38.214: Physical layer procedures for data," 3rd Generation Partnership Project (3GPP), Technical Specification 38.214, 2023, available: 3GPP TS 38.214 Release 17. [Online]. Available: http://www.3gpp.org/DynaReport/38214.htm

[39] ——, "TS 38.331: NR; Radio Resource Control (RRC); Protocol specification," 3rd Generation Partnership Project (3GPP), Technical Specification 38.331, 2023, available: 3GPP TS 38.331 Release 17. [Online]. Available: http://www.3gpp.org/DynaReport/38331.htm

[40] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[41] V. Gabillon, M. Ghavamzadeh, and A. Lazaric, "Best arm identification: A unified approach to fixed budget and fixed confidence," in *Advances in Neural Information Processing Systems*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012. [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/8b0d268963dd0cfb808aac48a549829f-Paper.pdf

[42] C. Qi, K. Chen, O. A. Dobre, and G. Y. Li, "Hierarchical codebook-based multiuser beam training for millimeter wave massive mimo," *IEEE Transactions on Wireless Communications*, vol. 19, no. 12, pp. 8142–8152, 2020.

[43] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5g cellular: It will work!" *IEEE access*, vol. 1, pp. 335–349, 2013.

[44] S. Kalyanakrishnan, A. Tewari, P. Auer, and P. Stone, "Pac subset selection in stochastic multi-armed bandits." in *ICML*, vol. 12, Edinburgh,Scotland, 2012, pp. 655–662.

[45] O. Besbes, Y. Gur, and A. Zeevi, "Stochastic multi-armed-bandit problem with non-stationary rewards," *Advances in neural information processing systems*, vol. 27, pp. 1–9, 2014.

[46] D. Ghosh, M. K. Hanawal, and N. Zlatanov, "Learning optimal phase-shifts of holographic metasurface transceivers," *arXiv preprint arXiv:2301.03371*, 2022.

[47] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge university press, 2019, vol. 48.

[48] A. Maurer and M. Pontil, "Empirical bernstein bounds and sample variance penalization," *arXiv preprint arXiv:0907.3740*, 2009.

[49] C.-Y. Wei, Y.-T. Hong, and C.-J. Lu, "Tracking the best expert in non-stationary stochastic environments," *Advances in neural information processing systems*, vol. 29, pp. 1–9, 2016.

[50] J. B. Nathan Blinn, Matthieu Bloch, "mlcomm," https://github.com/nrb5089/mlcomm, 2024, accessed: 2024-08-26.

[51] C. Liu, L. Zhao, M. Li, and L. Yang, "Adaptive beam search for initial beam alignment in millimetre-wave communications," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 6, pp. 6801–6806, 2022.

[52] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *Journal of the American Statistical Association*, vol. 22, no. 158, pp. 209–212, 1927.

[53] N. Blinn and M. Bloch, "mmwave beam alignment using hierarchical codebooks and successive subtree elimination," *arXiv preprint arXiv:2209.02896*, 2022.

[54] A. Garivier and E. Kaufmann, "Nonasymptotic sequential tests for overlapping hypotheses applied to near-optimal arm identification in bandit models," *Sequential Analysis*, vol. 40, no. 1, pp. 61–96, 2021.