Adversarial Multi-armed Bandit for mmWave Beam Alignment with One-Bit Feedback

Irched Chafaa
ETIS, UMR 8051, Université Paris Seine, Université Cergy-Pontoise, ENSEA, CNRS
Cergy, France
L2S, CentraleSupélec, Université Paris-Saclay
Gif-sur-Yvette, France
irched.chafaa@ensea.fr

E. Veronica Belmega
ETIS, UMR 8051, Université Paris Seine, Université Cergy-Pontoise, ENSEA, CNRS
Cergy, France
belmega@ensea.fr

Mérouane Debbah
CentraleSupélec, Université Paris-Saclay
Gif-sur-Yvette, France
Mathematical and Algorithmic Sciences Lab, Huawei France R& D
Paris, France
merouane.debbah@centralesupelec.fr

ABSTRACT

To exploit the large bandwidth available in the millimeter wave spectrum, highly directional beams need to be employed to compensate for the severe pathloss incurred at high frequencies. As a result, the beams of both the transmitter and the receiver must be constantly aligned. In this paper, the beam alignment (BA) problem is formulated as an adversarial multi-armed bandit (MAB) problem, yielding to a distributed BA search between the transmitter and receiver. First, we analyze the optimal codebook size for the BA that reduces the search space while insuring good performance levels. Then, we propose to use the exponential weights algorithm at both the transmitter and the receiver to match their beams. Remarkably, our distributed algorithm relies on a single bit of feedback information and its performance is demonstrated via numerical results.

CCS CONCEPTS

• **Theory of computation** → *Online learning theory*; • **Applied computing** → *Telecommunications*;

KEYWORDS

mmWave, beam alignment, multi-armed bandit, exponential weights algorithm

ACM Reference Format:
Irched Chafaa, E. Veronica Belmega, and Mérouane Debbah. 2019. Adversarial Multi-armed Bandit for mmWave Beam Alignment with One-Bit Feedback. In *Proceedings of ACM Valuetools conference (ValueTools'19).* ACM, New York, NY, USA, 8 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Because of the congestion of the sub-6 GHz microwave spectrum, the millimeter wave (mmWave) band, ranging from 30 GHz to 300 GHz, has been considered as a promising solution for future wireless networks [16, 17] to achieve the high data rates required by the data-hungry applications.

Propagation at mmWave frequencies is characterized by high pathloss caused by free-space pathloss [5], penetration loss and absorption by different components of the wireless environment [9, 18]. This suggests the use of highly directional beams by using large antenna arrays jointly with beamforming techniques [8, 13] to compensate for the high propagation loss. Luckily, the small wavelength of mmWave allows to place a large number of antenna elements in relatively small size arrays which yields a large beamforming gain [21] by focusing the signal's power toward the intended user's equipment (UE).

In order to ensure a reliable communication, the beams of the transmitter (Tx) and the receiver (Rx) have to be aligned. This problem is referred to as *initial beam training* or *beam alignment* (BA). The BA process must be operated before proceeding to data transmission to achieve the desired communication performance. An example that illustrates the critical role of BA is the experimental results in [15] that showed a 17 dB loss in the link budget for a 7° beam-width mmWave system caused by a mere misalignment of only 18°. Furthermore, BA is subject to a large overhead that grows fast with the resolution of the beam pattern.

Several schemes and algorithms have been proposed to perform BA and reduce the overall training overhead. The main idea consists of sampling the space in the angular domain and using a finite set of training vectors from a predefined codebook to scan the mmWave channel and align the transmitter and receiver beamforming and combining vectors, respectively. In the exhaustive search method of [12], the devices scan all possible beamformer-combiner pairs until they find the optimal one. The IEEE 802.11ad standard [11] proposed to narrow adaptively the beam-width according to a multi-level hierarchical scheme in order to reduce the training time. Another BA technique, with a wind induced beam misalignment analysis, was presented in [10] for analog beamforming systems using adaptive subspace sampling. The authors in [7] introduced a new protocol, based on the mmWave channel's sparsity, that uses multi-armed beams to cut the delay down to 2.5 ms for the implemented system. Based on the non-negative least squares (NNLS) technique, [21] presents another BA scheme to find the strongest path connecting each UE to the Tx.

ValueTools'19, March 2019, Palma de Mallorca, Spain

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

More recently, machine learning tools have been proposed to solve the BA problem. The authors in [2] proposed an online learning algorithm, called fast machine learning (FML), for mmWave vehicular communications. They modeled the BA problem as a multi-armed bandit (MAB) problem with contextual information (i.e., the vehicle's direction of arrival). In [6], the unimodal beam alignment (UBA) algorithm exploits the correlation between consecutive beam alignments (as contextual information) and the unimodality of the received power to reduce the search space and maximize the received energy. However, the architecture of these algorithms is centralized and the joint beamformer-combiner pair is simultaneously selected at the transmitter and receiver. This implies the existence of a central node in charge of selecting the beamformer-combiner pair, which requires heavy signaling feedback.

In this paper, we adopt an adversarial MAB formulation for the BA problem that allows us to decouple the problem and split the processing cost between the transmitter and receiver. Both nodes use a learning algorithm to choose their own beam direction, in a distributed manner without knowing each others choices. Hence, the learning is carried out at both the transmitter and receiver without relying on a central node as in [6] and [2]. An immediate advantage is that each node explores only its own set of beam directions and not the set of beam pairs which is much larger.

Because of the distributed setting, the stochastic MAB formulation no longer applies. Instead, we exploit the more general adversarial MAB setting [4]. Each node uses the exponential weights algorithm (EXP3) [3] to select a beamforming vector index (referred to as an action) and receives a reward (payoff) accordingly. The exponential weights algorithm provides an efficient solution to the distributed BA problem in terms of regret, which measures the performance gap with the best fixed solution on average. [3]. Moreover, the algorithm does not rely on any assumptions regarding the underlying dynamics of the system, which makes it relevant when the channel varies randomly because of the device mobility, the sudden blockage effect of the mmWave signal, etc. It also offers a good tradeoff between data exploration (trying different beam directions to find the best one) and exploitation (transmitting data over the beam directions believed to give optimal rewards).

Our main contributions can be summarized as follows. First, we start by investigating the optimal size of the beamforming codebook to be used for the BA in order to reduce the exploration cost. We provide the minimal codebook size that guarantees a certain quality of service (QoS) related to the outage probability. In the case of single path mmWave channels, a closed-form expression is obtained, while in multipath channels we use numerical experiments to compute this performance measure. We show that increasing the codebook resolution beyond a certain point does not offer a higher outage performance but implies an increasing exploration cost. Hence, restricting the number of beamforming vectors offers a good tradeoff between the outage probability and the exploration cost.

Second, we exploit the exponential weights algorithm for a distributed search of the best beam alignment in a point-to-point mmWave MIMO (Multiple Input Multiple Output) system. The performance of the proposed method is evaluated in terms of the notion of regret. Our simulation results show a decreasing average regret as the learning proceeds which implies more accuracy in the BA. The one-bit feedback proposed scheme allows the transmitter and the receiver to learn their best beamformer and combiner vectors that offer a good SNR level in a distributed manner and relying only on a single bit worth of feedback.

2 SYSTEM MODEL AND PROBLEM FORMULATION

We consider the mmWave MIMO system depicted in Figure 1. The transmitter (Tx) and the receiver (Rx) employ N_T and N_R RF chains connected to M_T and M_R antennas respectively such that $N_T \leq M_T$ and $N_R \leq M_R$. Both the transmitter and receiver use hybrid (analog and digital) codebooks $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_A]$ with $\mathbf{f}_i \in \mathbb{C}^{M_T}$ and $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_A]$ with $\mathbf{w}_j \in \mathbb{C}^{M_R}$, respectively. We assume the same number of possible vectors or codebook size $A = 2^n, n \in \mathbb{N}^\star$ at both Tx and Rx.

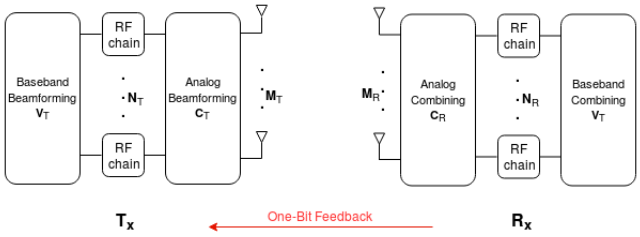

Figure 1: Point-to-point mmWave MIMO system.

The codebook $\mathbf{F} = \mathbf{C}_T\mathbf{V}_T$ is the combination of an analog beam-steering matrix $\mathbf{C}_T \in \mathbb{C}^{M_T \times N_T}$ and a digital beamformer $\mathbf{V}_T \in \mathbb{C}^{N_T \times A}$ as in [20]. The elements of $\mathbf{C}_T$ have constant modulus since it is obtained using phase shifters and each column is a unit norm beam-steering vector. The digital beamformer is normalized such that for each column vector $\mathbf{v}_T \in \mathbb{C}^{N_T}$ of $\mathbf{V}_T$, we have $\left\|\mathbf{C}_T\mathbf{v}_T\right\|_2^2 = 1$. The receiver codebook $\mathbf{W} = \mathbf{C}_R\mathbf{V}_R$ is constructed similarly.

The received signal for the beamforming-combining pair $(\mathbf{f}_i, \mathbf{w}_j)$ is given by:

$$y_{i,j} = \mathbf{w}_j^H \mathbf{H} \mathbf{f}_i x + \mathbf{w}_j^H \mathbf{n}, \tag{1}$$

with $\mathbf{H} \in \mathbb{C}^{M_R \times M_T}$ the channel matrix; $x \in \mathbb{C}$ the transmitted pilot symbol such that $\mathbb{E}[\mid x \mid^2] = P_T$ where P_T is the transmitting power during the BA; $\mathbf{n} \sim \mathcal{N}(0, \sigma_n^2)$ the Gaussian noise vector.

Given the sparse nature of the mmWave channel [8, 14, 19], we use the geometric channel model with L scatterers where each scatterer contributes with a single propagation path. The channel matrix is given by [1]:

$$\mathbf{H} = \sqrt{\frac{M_T M_R}{\rho}} \sum_{l=1}^{L} \alpha_\ell \mathbf{a}_R(\theta_\ell) \mathbf{a}_T(\phi_\ell)^H, \tag{2}$$

where ρ represents the average pathloss that depends on the carrier frequency, the Tx-Rx distance and the propagation environment [18]; $\alpha_\ell \sim \mathcal{N}(0, \sigma_{\alpha_\ell}^2)$, $\ell \in \{1, 2, .., L\}$ is the complex path's gain assumed to be Gaussian distributed; σ_{α_ℓ} is the average power gain. For each path, the azimuth angles of departure and arrival (AoDs/AoDs)

are denoted by $\phi_\ell \in [0, 2\pi]$ and $\theta_\ell \in [0, 2\pi]$. The vectors $\mathbf{a}_R(\theta)$ and $\mathbf{a}_T(\phi)$ represent the array response vectors for the receiver and the transmitter. When using a uniform linear array (ULA), they are defined as:

$$\mathbf{a}_R(\theta_\ell) = \frac{1}{\sqrt{M_R}}[1, e^{j\frac{2\pi}{\lambda}d\sin(\theta_\ell)}, ..., e^{j(M_R-1)\frac{2\pi}{\lambda}d\sin(\theta_\ell)}]^T, \quad (3)$$

$$\mathbf{a}_T(\phi_\ell) = \frac{1}{\sqrt{M_T}}[1, e^{j\frac{2\pi}{\lambda}d\sin(\phi_\ell)}, ..., e^{j(M_T-1)\frac{2\pi}{\lambda}d\sin(\phi_\ell)}]^T, \quad (4)$$

where d represents the distance between the array elements. The resulting signal-to-noise ratio (SNR) at the receiver can be written as:

$$SNR_{i,j} = \frac{|\ \mathbf{w}_j^H \mathbf{H}\mathbf{f}_i\ |^2\ P_T}{\sigma_n^2}. \quad (5)$$

Problem Formulation

In MAB problems, the objective of the nodes is usually to maximize their mean rewards. In our scheme, we assume a binary reward that equals one when the SNR at the receiver is higher than a threshold ξ and zero otherwise:

$$r(i,j) = \begin{cases} 1, & \text{if} \quad SNR_{i,j} \geqslant \xi, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

The threshold ξ represents the minimum SNR needed to have a reliable link between Tx and Rx and has to be carefully chosen. The mean reward $\mathbb{E}[r(i,j)]$ can be expressed as:

$$\begin{aligned} \mathbb{E}[r(i,j)] &= 0 * \mathbf{P}[SNR_{i,j} < \xi] + 1 * \mathbf{P}[SNR_{i,j} \geqslant \xi] \\ &= \mathbf{P}[SNR_{i,j} \geqslant \xi], \end{aligned} \quad (7)$$

where the expectation $\mathbb{E}[.]$ is taken over the randomness of the channel.

In this work, we aim at minimizing an outage probability $\mathbf{P}_{out}$ that we define as:

$$\mathbf{P}_{out} \triangleq \mathbf{P}[SNR_{i,j} < \xi]. \quad (8)$$

From (7) and (8), we can see that minimizing the outage probability $\mathbf{P}_{out}$ is equivalent to maximizing the mean reward $\mathbb{E}[r(i,j)]$, which is the common assumption in general MAB problems. Remark that the outage probability in (8) depends on the codebook size as larger codebooks provide higher beamforming gains and, hence, higher SNRs.

We define ΔSNR to quantify how much we should increase the codebook size A and still get a significant performance improvement in terms of the SNR. We use this measure as a criterion to avoid increasing the codebook size uselessly at the expense of larger exploration duration.

$$\Delta SNR = SNR_0 - \max_{k \in \{1,2,...,A^2\}} SNR_k, \quad (9)$$

where the index k refers to the indices of all possible pairs (i,j) such that $k \in \{1, 2, .., A^2\}$ and $SNR_k = SNR_{i,j}$; SNR_0 represents the highest possible SNR related to the channel conditions and independent from the used beamforming codebooks such that:

$$SNR_0 = \max_{\mathbf{u},\mathbf{v}} \frac{|\mathbf{u}^H \mathbf{H}\mathbf{v}|^2 P_T}{\sigma_n^2} = \frac{|\mathbf{u}_o^H \mathbf{H}\mathbf{v}_o|^2 P_T}{\sigma_n^2}, \quad (10)$$

where $\mathbf{u}$ and $\mathbf{v}$ are the left-singular vectors and right-singular vectors of $\mathbf{H}$ respectively; $\mathbf{u}_o$ and $\mathbf{v}_o$ are the singular vectors corresponding to the largest singular value of $\mathbf{H}$.

To avoid having a central node that makes all the decisions, we consider that both Tx and Rx use a learning algorithm within a distributed architecture. As a result, there are only A actions to choose from for each node instead of A^2 beamformer-combiner pairs, as in the centralized setting [6]. Because of this, we can no longer use the stochastic MAB formulation, but instead we use the adversarial MAB formulation.

In the adversarial MAB context, each node $q \in \{T, R\}$ selects a beam direction (index of the beamforming vector) which is referred to as an action $s(t), s \in \mathcal{S}$ with $\mathcal{S} = \{1, 2, .., A\}$ at round $t \in \{1, 2, ..., \mathcal{T}\}$. A reward $r(s(t))$ is assigned to the action depending on the corresponding SNR at the receiver.

At the transmitter side.
The transmitter searches for the beamforming vector index $i \in \mathcal{S}$ that minimizes the outage probability $\mathbf{P}_{out}$ without knowing the Rx policy $j(t) \in \mathcal{S}$ at round $t \in \{1, 2, ..., \mathcal{T}\}$ and $\mathcal{T} \leq T_c$ with T_c the channel coherence time. Hence, Tx aims at solving the following online optimization problem:

$$\begin{aligned} \forall t, \quad \min_{i} \quad & \mathbf{P}_{out}\ (i, j(t)) \\ \text{s.t.} \quad & i \in \mathcal{S}. \end{aligned} \quad (11)$$

Since the strategy of the receiver cannot be anticipated, Tx has to solve (11) with an unknown objective. Instead, Tx uses an online algorithm based on strictly causal feedback from Rx. To evaluate the performance of the online algorithm, we use the notion of average regret [4], which evaluates the performance gap between the proposed strategy and a fixed one, which is the best BA strategy on average over the time horizon. We denote the online policy of Tx by $\{s_T(t)\}_{t=1,..,\mathcal{T}}$ and the average regret expression for Tx can be written as:

$$Reg_T = \frac{1}{\mathcal{T}} \left(\max_i \sum_{t=1}^{\mathcal{T}} r\ (i, j(t)) - \sum_{t=1}^{\mathcal{T}} r\ (s_T(t), j(t)) \right). \quad (12)$$

At the receiver side.
Similarly, the receiver wishes to find the combining vector index j that minimizes $\mathbf{P}_{out}$ without any knowledge of the choice of the beamforming index at the transmitter $i(t) \in \mathcal{S}$ at round t. Thus, we have the following online optimization problem at the Rx node:

$$\begin{aligned} \forall t, \quad \min_{j} \quad & \mathbf{P}_{out}\ (i(t), j) \\ \text{s.t.} \quad & j \in \mathcal{S}, \end{aligned} \quad (13)$$

which also has an unknown objective function. An online algorithm is also used at Rx to find the optimal combiner. We denote the online policy of Rx by $\{s_R(t)\}_{t=1,..,\mathcal{T}}$ and the average regret incurred by the Rx online policy is written as:

$$Reg_R = \frac{1}{\mathcal{T}} \left(\max_j \sum_{t=1}^{\mathcal{T}} r\ (i(t), j) - \sum_{t=1}^{\mathcal{T}} r\ (i(t), s_R(t)) \right). \quad (14)$$

One of the main objectives in this work is to find online and distributed policies at both Tx and Rx that minimize the average regret and asymptotically lead to no regret when $\mathcal{T} \to +\infty$; all this while relying on a single bit of feedback information. This means that the online policies will perform on average at least as well as the optimal fixed policies.

3 OPTIMAL CODEBOOK SIZE

Before presenting the BA online policies, we discuss the optimal codebook size (or spatial resolution). We assume that the beam directions of the codebook are uniformly chosen to cover the spatial horizon between $-\pi/2$ and $\pi/2$ by dividing the angular domain by half each time we increase the codebook size. The higher the codebook size, the narrower and more directed the beams are. Theoretically at least, as long as we increase the codebook size the received SNR also increases. The downside is that the set of candidate beamforming vectors increases, which implies a higher exploration cost. The rising question is then: *what is the codebook size that balances best the received SNR and the exploration cost?*

In order to optimize the codebook size, we first analyze the probability $\mathbf{P}(\Delta SNR \leq \epsilon)$ where ϵ represents the maximum allowed gap between SNR_0 and $\max_k \ SNR_k$. Then, we search for the smallest codebook size (for small exploration costs) which provides a high enough value of this probability (for high performance in terms of SNR) using numerical simulations.

3.1 Single path case: $L = 1$

For simplicity, we start with the single path mmWave channel, in which $\mathbf{H} = \sqrt{\frac{M_T M_R}{\rho}} \alpha \ \mathbf{a}_R(\theta) \ \mathbf{a}_T(\phi)^H$. We can express SNR_0 and SNR_k as:

$$SNR_0 = \frac{P_T M_T M_R}{\rho \sigma_n^2} \ |\mathbf{u}_o^H \mathbf{a}_R(\theta) \ \mathbf{a}_T(\phi)^H \ \mathbf{v}_o|^2 \ |\alpha|^2 = B \, C_0 \ |\alpha|^2, \quad (15)$$

$$SNR_k = \frac{P_T M_T M_R}{\rho \sigma_n^2} \ |\mathbf{w}_j^H \ \mathbf{a}_R(\theta) \, \mathbf{a}_T(\phi)^H \ \mathbf{f}_i|^2 \ |\alpha|^2 = B \, C_k \ |\alpha|^2, \quad (16)$$

where $B = \frac{P_T M_T M_R}{\rho \sigma_n^2}$, $C_0 = |\mathbf{u}_o^H \mathbf{a}_R(\theta) \ \mathbf{a}_T(\phi)^H \ \mathbf{v}_o|^2$ and $C_k = |\mathbf{w}_j^H \ \mathbf{a}_R(\theta) \, \mathbf{a}_T(\phi)^H \ \mathbf{f}_i|^2$.

Thus, we can write ΔSNR as:

$$\Delta SNR = B \ |\alpha|^2 \left(C_0 - \max_k C_k \right). \quad (17)$$

Assuming that the path gain follows complex Gaussian distribution, while the other parameters are fixed and deterministic, then its amplitude squared follows an exponential distribution, i.e, $|\alpha|^2 \sim Exp\ (\frac{1}{\sigma_\alpha})$ which results in an exponential distribution of ΔSNR such that $\Delta SNR \sim Exp\left(\frac{1}{\sigma_\alpha \ B \left(C_0 - \max_k C_k \right)} \right)$. Hence, we can find a closed-form expression of $\mathbf{P}(\Delta SNR \leq \epsilon)$ as:

$$\mathbf{P}(\Delta SNR \leq \epsilon) = 1 - \exp\left(\frac{-\epsilon}{\sigma_\alpha \ B \left(C_0 - \max_k C_k \right)} \right). \quad (18)$$

When the noise variance is very small ($\sigma_n^2 \to 0$), this probability approaches zero and the obtained SNR approaches the ideal value SNR_0. When the noise variance is very large ($\sigma_n^2 \to +\infty$), the obtained SNR cannot approach this ideal value.

3.2 Multipath case: $L > 1$

In the more general multipath mmWave channel case, we can express SNR_0 and SNR_k as follows:

$$SNR_0 = B \left| \sum_{\ell=1}^{L} \alpha_\ell D_\ell \right|^2 = B \, Z \quad (19)$$

$$SNR_k = B \left| \sum_{\ell=1}^{L} \alpha_\ell D_{\ell k} \right|^2 = B \, J_k, \quad (20)$$

where Z and J_k are defined as $Z = \left| \sum_{\ell=1}^{L} \alpha_\ell D_\ell \right|^2$ and $J_k = \left| \sum_{\ell=1}^{L} \alpha_\ell D_{\ell k} \right|^2$ for $k \in \{1, 2, ..., A^2\}$; $D_\ell = \mathbf{u}_o^H \mathbf{a}_R(\theta_\ell) \ \mathbf{a}_T(\phi_\ell)^H \mathbf{v}_o$ and $D_{\ell k} = \mathbf{w_j}^H \mathbf{a}_R(\theta_\ell) \ \mathbf{a}_T(\phi_\ell)^H \mathbf{f}_i$.

Assuming the path gains α_ℓ, $\ell \in \{1, 2, .., L\}$ to be Gaussian distributed, the sums $\sum_{\ell=1}^{L} \alpha_\ell D_\ell$ and $\sum_{\ell=1}^{L} \alpha_\ell D_{\ell k}$ follow the Gaussian distributions $\mathcal{N}\left(0, \sum_{\ell=1}^{L} \sigma_{\alpha_\ell} |C_\ell|^2\right)$ and $\mathcal{N}\left(0, \sum_{\ell=1}^{L} \sigma_{\alpha_\ell} |C_{\ell k}|^2\right)$, respectively. Therefore, the random variables Z and J_k follow an exponential distribution such that $Z \sim Exp\left(\frac{1}{\sum_{\ell=1}^{L} \sigma_{\alpha_\ell} |C_\ell|^2} \right)$ and $J_k \sim Exp\left(\frac{1}{\sum_{\ell=1}^{L} \sigma_{\alpha_\ell} |C_{\ell k}|^2} \right)$.

Using (19) and (20), we obtain:

$$\Delta SNR = B \left(Z - \max_k J_k \right). \quad (21)$$

We denote $X = \max_k \ J_k$ the maximum value of A^2 exponential random variables: $J_1, J_2, ..., J_{A^2}$. Since both Z and X depend on the path gains, they are correlated random variables. Finding the joint distribution of Z and X is non trivial and, hence, we cannot obtain a closed-form expression of the distribution of ΔSNR. Consequently and as opposed to the particular single path case, we can only compute $\mathbf{P}(\Delta SNR \leq \epsilon)$ empirically via Monte-Carlo simulations.

3.3 Numerical Results

Having analyzed $\mathbf{P}(\Delta SNR \leq \epsilon)$, we will use numerical simulations to find a good tradeoff between a small codebook size and a good performance in terms of ΔSNR.

In Figure 2, we evaluate the probability $\mathbf{P}(\Delta SNR \leq \epsilon)$ as a function of $\log_2(A)$, where A is the codebook size, for the scenario: $M_T = 32$, $M_R = 4$, $N_T = 4$, $N_R = 2$, $P_T = 30$ dBm, $\sigma_{\alpha_\ell} = 1$, at 28 GHz carrier frequency and 1 GHz signal bandwidth. Both the transmitter and the receiver are equipped with ULAs of $\lambda/2$ spacing between the array elements. The pathloss ρ is calculated as in [18, equation (2)]. The empirical results, for the multipath case ($L = 3$), are obtained using Monte-Carlo simulations with 100,000 independent channel realizations, whereas for the single path case ($L = 1$) the closed-form expression is used.

We notice that the highest SNR obtained by the codebook is more probable to approach SNR_0 when the codebook size increases. In fact, as the size A becomes larger, the beams tend to be narrower which allows for a better alignment with the channel's best spatial path. Moreover, as the parameter ϵ becomes smaller, the necessary codebook size to reach high probability becomes larger.

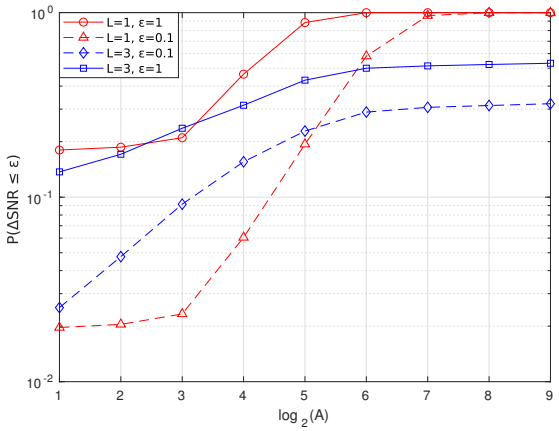

Figure 2: $\mathbf{P}(\Delta SNR \leq \epsilon)$ as a function of the codebook size A for the single path and multipath channels. Increasing the codebook size beyond a certain value, does not bring a significant performance improvement.

Nevertheless, we notice that ever increasing the codebook size, does not lead to a significant increase in $\mathbf{P}(\Delta SNR \leq \epsilon)$. For instance, in the case of a single path and $\epsilon = 1$, the gain in the probability is 0.4 when we move from $A = 16$ to $A = 32$ while it is only 0.12 when we increase the size from $A = 32$ to $A = 64$. We conclude that we don't need to continuously keep increasing the codebook size to get significantly higher SNR at the receiver. We can limit the number of beamforming vectors to reduce the BA duration. For the proposed BA scheme in the next section, we use the size $A = 32$ for the single path channel and $A = 64$ for the multipath channel.

4 BEAM ALIGNMENT USING EXPONENTIAL WEIGHTS ALGORITHM

In order to establish a reliable connection, the transmitter and the receiver need to align their beams as best as possible in order to guarantee the required QoS. To this aim, we exploit the exponential weights algorithm (**Exp3**), which is a no-regret learning algorithm introduced originally for the adversarial MAB problem in [3].

The basic idea of the exponential weights algorithm is as follows. At each round t, node $q \in \{T, R\}$ selects an action $s_q(t)$ with a probability $p_{s_q}(t)$ that is exponentially proportional to the total reward scored by that action (until round t): $G_{s_q}(t) = \sum_{\tau=1}^{t} r\left(s_q(\tau)\right)$. This means that the actions providing high cumulative rewards in the past become more probable to be chosen. In our case, the algorithm explores different beam directions to identify the best one to be exploited for data transmission. The more the exploration phase lasts the more informed the nodes become, but at the expense of under-exploiting the already available good directions (exploration-exploitation tradeoff). The details of the beam alignment scheme are given next.

4.1 Learning at the transmitter

At round t, the Tx chooses an action (beamforming vector index) $s_T(t) \in \{1, \ldots, A\}$ according to the probability distribution $p_T(t) = (p_{T,1}, \ldots, p_{T,A})$ such that:

$$p_{T,i}(t) = (1-\gamma)\frac{\exp(\eta G_i(t-1))}{\sum_{v_T=1}^{A}\exp(\eta G_{v_T}(t-1))} + \frac{\gamma}{A}, \quad \forall i \in \{1, \ldots, A\} \tag{22}$$

where $\eta > 0$ and $\gamma \in [0, 1]$ are the learning parameters.

This means that Tx chooses a beam direction and then transmits its symbol. The reward $r(s_T(t))$ for the chosen beam direction $s_T(t)$ is given as:

$$r(s_T(t)) = \begin{cases} 1, & if \quad SNR_{s_T(t),j(t)} \geq \xi, \\ 0 & otherwise. \end{cases} \tag{23}$$

All the other directions (not chosen) are assumed to have a zero reward at round t.

Since the transmitter does not know the policy $j(t)$ and can not compute the receiver SNR at round t, instead we assume that receiver sends a one-bit worth of feedback at each round representing precisely whether the incurred SNR was below or above the chosen threshold. Finally, the cumulative rewards $G_i(t), \forall i$ and the probability distribution $p_T(t)$ are updated, and the process repeats. The transmitter's BA scheme is summarized in the algorithm **BA-Tx** below.

BA-Tx: Exponential Weights for Beam Alignment at Tx

Parameters $\eta > 0$ and $\gamma \in [0, 1]$
Initialization: Set $G_i(0) = 0$ and $p_{T,i}(0) = 1/A, \forall i$.
Repeat for $t = 1, 2, \ldots, \mathcal{T}$
  Select action $s_T(t)$ with probability distribution $p_T(t)$.
  Update the reward $r(s_T(t))$.
  Update the cumulative rewards:

$$\begin{cases} G_{s_T}(t) = G_{s_T}(t-1) + r(s_T(t)), \\ G_i(t) = G_i(t-1), \forall\, i \neq s_T(t). \end{cases}$$

  Update the distribution $p_T(t)$ as in (22).

4.2 Learning at the receiver

Similarly to the transmitter, Rx selects a combiner index $s_R(t) \in \{1, \ldots, A\}$ at round t according to its own probability distribution $p_R(t) = (p_{R,1}(t), \ldots, p_{R,A}(t))$ such that:

$$p_{R,j}(t) = (1-\gamma)\frac{\exp(\eta G_j(t-1))}{\sum_{v_R=1}^{A}\exp(\eta G_{v_R}(t-1))} + \frac{\gamma}{A}, \quad \forall j \in \{1, \ldots, A\} \tag{24}$$

Depending on the obtained SNR at the receiver, the chosen action $s_R(t)$ incurs the reward $r(s_R(t))$ with:

$$r(s_R(t)) = \begin{cases} 1, & if \quad SNR_{i(t),s_R(t)} \geqslant \xi, \\ 0 & otherwise, \end{cases} \tag{25}$$

while the other actions (not chosen) receive a zero reward.

Once the received SNR value is estimated, Rx sends a one-bit feedback to the Tx, which is simply the obtained reward $r(s_R(t))$ at round t. The value of this feedback (one or zero) determines the reward $r(s_T(t))$ of the chosen action $s_T(t)$ at the Tx node.

At the end of each round, the Rx updates both the cumulative rewards $G_j(t), \forall j$ and the probability distribution $p_R(t)$ to be used for the next round. The different steps of the receiver's scheme are outlined in the **BA-Rx** algorithm.

BA-Rx: Exponential Weights for Beam Alignment at Rx

Parameters $\eta > 0$, $\gamma \in [0,1]$, ξ
Initialization: Set $G_j(0) = 0$ and $p_{R,j}(0) = 1/A$, $\forall j$.
Repeat for $t = 1, 2, ..., \mathcal{T}$
- Select action $s_R(t)$ with probability distribution $p_R(t)$.
- Update the reward $r(s_R(t))$.
- Send feedback to Tx.
- Update the cumulative rewards:
$$\begin{cases} G_{s_R}(t) = G_{s_R}(t-1) + r(s_R(t)), \\ G_j(t) = G_j(t-1), \forall\, j \neq s_R(t). \end{cases}$$
- Update the distribution $p_R(t)$ as in (24) .

The next result follows from [3].

Corollary 4.1. *If the rewards are in the range* $[0,1]$ *and the exponential weights algorithm is run with parameters* $\eta = \frac{\gamma}{A}$ *and* $\gamma = min\left\{1, \sqrt{\frac{A \log A}{(e-1)\mathcal{T}}}\right\}$, *then the expected regret of the algorithm is bounded as:*

$$Reg_q \leq 2\sqrt{e-1}\sqrt{\frac{A \log A}{\mathcal{T}}}, \tag{26}$$

with $q \in \{T, R\}$ *representing the transmitter and receiver nodes and* $e = \exp(1)$.

The upper bound of the regret in (26) implies a certain guarantee on the BA accuracy at the nodes running the exponential weights algorithm with an appropriate choice of the parameters η and γ. When the time horizon is large, the online algorithms perform at least as well as the respective average optimal solutions (at Tx and Rx). We also see that the average regret decays to zero as $O(1/\sqrt{\mathcal{T}})$.

4.3 Numerical Results

In this subsection, we analyze the proposed BA algorithms using a codebook size of either $A = 32$ or $A = 64$ (depending on the number of path components L), which is a good tradeoff between beamforming accuracy and exploration cost as we discussed in Section 3.

The simulation setup and the parameters of the mmWave system are the same as in Subsection 3.3 unless stated otherwise. The hybrid codebook used for the simulations is generated offline using [20, Algorithm 1] because it supports the mmWave hybrid architecture and performs well in terms of beamforming gain compared to other existing codebooks in the literature. Also, the parameter ξ in (23) and (25) is set at 6 dB.

Average regret performance.
We assume that both nodes (Tx and Rx) run their own BA algorithms in a distributed manner. In Figure 3, we illustrate the average regret at the transmitter side Reg_T and compare the BA-Tx algorithm with a naive beam-selection procedure consisting of choosing a random direction following the uniform distribution. The codebook size is $A = 32$ for a single path channel ($L = 1$) and the curves are averaged over 10,000 independent channel realizations.

We remark that the regret decays to zero when using the exponential learning algorithms BA-Tx and BA-Rx and remains below the upper-bound in (26). On the other hand, the naive beam-selection procedure performs very poorly in terms of regret.

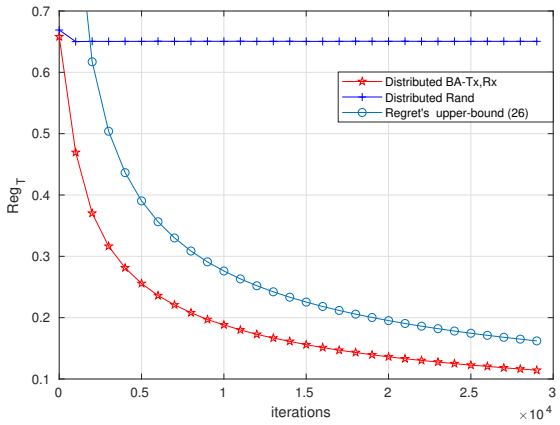

Figure 3: Average regret at the transmitter as a function of the number of iterations. The exponential learning leads to no regret, while the naive random beam-selection clearly does not.

Impact of the codebook size.
We now evaluate the impact of the codebook size on the regret performance. Figure 4 depicts the average regret at the transmitter after $\mathcal{T} = 10,000$ iterations of the BA-Tx and BA-Rx algorithms. The curves are averaged over 10,000 independent channel realizations.

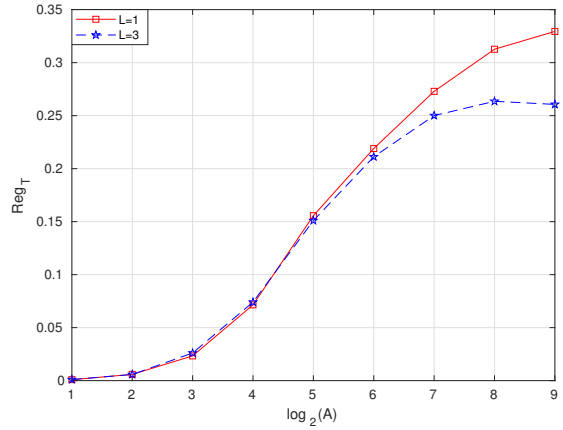

Figure 4: Transmitter's average regret Reg_T for different codebook sizes. For a fixed horizon $\mathcal{T}$, the average regret increases with the codebook size; when the set of possible beam directions increases, learning the best one takes longer.

We notice that the average regret increases as the codebook size becomes larger. Indeed, the algorithms explore a higher number of beam directions when the codebook size increases. This fact strengthens our discussion in Section 3 and the results of Figure 2; it is not practical to keep increasing the codebook size since the SNR improvement becomes marginal and the data exploration cost increases.

Beam alignment performance.
We have seen that the exponential learning algorithms BA-Tx and BA-Rx provide good performance in terms of regret. Nevertheless, our main objective is to propose beam alignment techniques that minimize the outage probability defined in (8). For this, we illustrate the outage probability in Figure 5. Our distributed algorithms are compared with the naive random scheme described above and with a centralized scheme, in which a similar exponential learning procedure is applied but over the set of A^2 beamforming-combining pairs. The outage probability is computed over 10, 000 independent channel realizations with a codebook size $A = 32$ for single path channels ($L = 1$) and a codebook size $A = 64$ for multipath channels ($L = 3$).

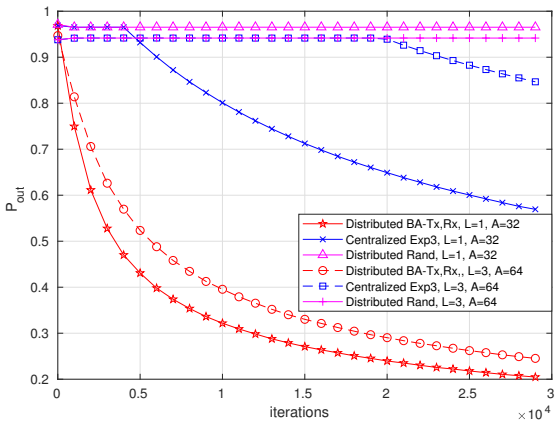

Figure 5: Outage probability P_{out} as a function of the number of iterations. Our distributed beam alignment scheme outperforms its centralized counterpart and the naive random beam-selection.

It can be seen that our distributed beam alignment scheme performs better than the random beam-selection and the centralized exponential learning of the joint beamforming-combining vectors. In mmWave channels, the beams at the transmitter have to be accurately aligned with the ones at the receiver, hence, randomly choosing the beams independently at the transmitter and the receiver will result in unaligned beams and a poor outage probability. Regarding the centralized counterpart, the joint selection increases the exploration set to A^2 possible pairs instead of A elements, which explains the worse performance compared with the distributed scheme.

In Figure 6, we consider the latency-reliability tradeoff as we evaluate the minimum number of iterations for the received SNR to reach the threshold ξ. The results are averaged over 100, 000 independent channel realizations. Figure 6 shows that increasing the SNR threshold ξ requires exploring more directions. This leads to a tradeoff between the exploration cost (which impacts the system latency) and the reliability (QoS) of the established link between Tx and Rx.

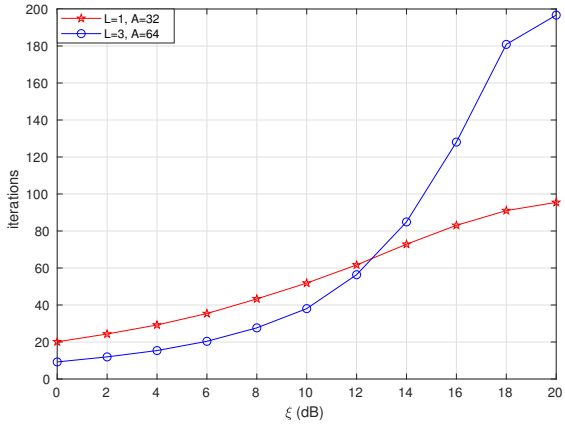

Figure 6: Minimum number of iterations to reach the SNR threshold ξ as a function of ξ. The exploration is longer to reach higher thresholds.

5 CONCLUSIONS

In this paper, we have addressed the beam alignment problem in a point-to-point mmWave MIMO system by using the adversarial multi-armed bandit problem and proposed a distributed alignment scheme, in which the transmitter and the receiver choose their beamforming and combining vectors independently from each other. We have also investigated the optimal codebook size that best trades off the received SNR for low beam exploration cost. Our beam alignment scheme is based on the exponential weights algorithm and is shown to provide promising performance in terms of outage probability. Future work may consider including contextual information to reduce the search set.

REFERENCES

[1] Ahmed Alkhateeb, Omar El Ayach, Geert Leus, and Robert W Heath. 2014. Channel estimation and hybrid precoding for millimeter wave cellular systems. *IEEE Journal of Selected Topics in Signal Processing* 8, 5 (2014), 831–846.
[2] Arash Asadi, Sabrina Müller, Gek Hong Allyson Sim, Anja Klein, and Matthias Hollick. 2018. FML: Fast Machine Learning for 5G mmWave Vehicular Communications. In *2018 IEEE International Conference on Computer Communications (INFOCOM)*.
[3] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. 1995. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. IEEE, 322–331.
[4] E Veronica Belmega, Panayotis Mertikopoulos, Romain Negrel, and Luca Sanguinetti. 2018. Online convex optimization and no-regret learning: Algorithms, guarantees and applications. *arXiv preprint arXiv:1804.04529* (2018).
[5] Harald T Friis. 1946. A note on a simple transmission formula. *Proceedings of the IRE* 34, 5 (1946), 254–256.
[6] Morteza Hashemi, Ashu Sabharwal, C Emre Koksal, and Ness B Shroff. 2017. Efficient Beam Alignment in Millimeter Wave Systems Using Contextual Bandits. *arXiv preprint arXiv:1712.00702* (2017).
[7] Haitham Hassanieh, Omid Abari, Michael Rodreguez, Mohammed Abdelghany, Dina Katabi, and Piotr Indyk. 2017. Agile millimeter wave networks with provable guarantees. *arXiv preprint arXiv:1706.06935* (2017).
[8] Robert W Heath, Nuria Gonzalez-Prelcic, Sundeep Rangan, Wonil Roh, and Akbar M Sayeed. 2016. An overview of signal processing techniques for millimeter wave MIMO systems. *IEEE Journal of Selected Topics in Signal Processing* 10, 3 (2016), 436–453.
[9] Ibrahim Hemadeh, Katla Satyanarayana, Mohammed El-Hajjar, and Lajos Hanzo. 2017. Millimeter-wave communications: Physical channel models, design considerations, antenna constructions and link-budget. *IEEE Communications Surveys & Tutorials* (2017).

[10] Sooyoung Hur, Taejoon Kim, David J Love, James V Krogmeier, Timothy A Thomas, and Amitava Ghosh. 2013. Millimeter wave beamforming for wireless backhaul and access in small cell networks. *IEEE Transactions on Communications* 61, 10 (2013), 4391–4403.

[11] Part 11 IEEE P802.11 ad. 2012. Wireless lan medium access control (MAC) and physical layer (PHY) specifications amendment 3: Enhancements for very high throughput in the 60 GHz band. *IEEE Computer Society* (2012).

[12] Cheol Jeong, Jeongho Park, and Hyunkyu Yu. 2015. Random access in millimeter-wave beamforming cellular networks: Issues and approaches. *IEEE Communications Magazine* 53, 1 (2015), 180–185.

[13] Andreas F Molisch, Vishnu V Ratnam, Shengqian Han, Zheda Li, Sinh Le Hong Nguyen, Linsheng Li, and Katsuyuki Haneda. 2017. Hybrid beamforming for massive MIMO: A survey. *IEEE Communications Magazine* 55, 9 (2017), 134–141.

[14] James N Murdock, Eshar Ben-Dor, Yijun Qiao, Jonathan I Tamir, and Theodore S Rappaport. 2012. A 38 GHz cellular outage study for an urban outdoor campus environment. In *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*. IEEE, 3085–3090.

[15] Thomas Nitsche, Adriana B Flores, Edward W Knightly, and Joerg Widmer. 2015. Steering with eyes closed: mmWave beam steering without in-band measurement. In *2015 IEEE International Conference on Computer Communications (INFOCOM)*. IEEE, 2416–2424.

[16] Yong Niu, Yong Li, Depeng Jin, Li Su, and Athanasios V Vasilakos. 2015. A survey of millimeter wave communications (mmWave) for 5G: Opportunities and challenges. *Wireless Networks* 21, 8 (2015), 2657–2676.

[17] Theodore S Rappaport, Shu Sun, Rimma Mayzus, Hang Zhao, Yaniv Azar, Kevin Wang, George N Wong, Jocelyn K Schulz, Mathew Samimi, and Felix Gutierrez. 2013. Millimeter wave mobile communications for 5G cellular: It will work! *IEEE access* 1 (2013), 335–349.

[18] Theodore S Rappaport, Yunchou Xing, George R MacCartney, Andreas F Molisch, Evangelos Mellios, and Jianhua Zhang. 2017. Overview of Millimeter Wave Communications for Fifth-Generation 5G Wireless Networks With a Focus on Propagation Models. *IEEE Transactions on Antennas and Propagation* 65, 12 (2017), 6213–6230.

[19] Akbar M Sayeed and Vasanthan Raghavan. 2007. Maximizing MIMO capacity in sparse multipath with reconfigurable antenna arrays. *IEEE Journal of Selected Topics in Signal Processing* 1, 1 (2007), 156–166.

[20] Jiho Song, Junil Choi, and David J Love. 2015. Codebook design for hybrid beamforming in millimeter wave systems. In *2015 IEEE International Conference on Communications (ICC)*. IEEE, 1298–1303.

[21] Xiaoshen Song, Saeid Haghighatshoar, and Giuseppe Caire. 2018. Efficient Beam Alignment for mmWave Single-Carrier Systems with Hybrid MIMO Transceivers. *arXiv preprint arXiv:1806.06425* (2018).