

Millimeter Wave Beam Recommendation via Tensor Completion

Tzu-Hsuan Chou, Nicolo Michelusi, David J. Love, and James V. Krogmeier

Abstract—Accurate and fast beam-alignment is essential to cope with the fast-varying environment in millimeter-wave communications. A data-driven approach is a promising solution to reduce the training overhead by leveraging side information and on-the-field measurements. In this work, a two-stage tensor completion algorithm is proposed to predict the received power on a set of possible users' positions, given received power measurements on a small subset of positions. Based on these predictions and on positional side information, a small subset of beams is recommended to reduce the training overhead of beam-alignment. Numerical results evaluated with the Quadriga channel simulator demonstrate that the proposed algorithm achieves correct alignment with high probability using small training overhead: given power measurement on only 20% of the possible positions when using a discrete coverage area, our algorithm attains a probability of correct alignment of 80%, with only 2% of trained beams, as opposed to a state-of-the-art scheme which achieves 50% correct alignment in the same configuration. To the best of our knowledge, this is the first work to consider the beam recommendation problem based on measurements collected on a small subset of positions.

Index Terms—Millimeter wave, beam-alignment, position-aided, tensor completion, sparse learning.

I. INTRODUCTION

Millimeter wave (mmWave) and massive MIMO are the key technologies to enable high throughput communication in future wireless systems, with applications such as video-streaming, automated driving, cloud computing, etc [1]–[4]. However, narrow beams are required to compensate the path loss and severe signal propagation at the mmWave frequencies. Narrow beam communication is especially challenging in mobile environments, since the beam direction needs to be continuously trained. Typically, this is achieved by sweeping over a finite set of candidate beamforming vectors to find the strongest beam direction [5], [6]. This process incurs huge overhead [7] due to the potentially large set of candidate beamforming solutions that should be searched for in large antenna systems, calling for efficient beam-alignment protocols [8].

Beam-alignment has been a subject of intense research in recent years, with techniques ranging from beam-sweeping [8], angle of arrival and of departure (AoA/AoD) estimation [9], to data-assisted schemes [5]. In particular, beam-sweeping schemes require to collect a set of beam measurements over the entire beam-space. The simplest form of beam-sweeping is *exhaustive search*, which scans through all possible beams between transmitter and receiver. AoA/AoD estimation reduces the number of measurements by leveraging the sparsity of mmWave channels via compressive sensing [9]. In data-assisted schemes, mmWave channels are related to the environment of the user, such as its position, the geometry of the surrounding environment (e.g., buildings, vegetation,

etc.) or temporal information (e.g., traffic). Due to the difficulty to comprehensively model and accurately represent all propagation features in the environment, and how these affect mmWave propagation, a data-driven approach based on machine learning may be envisioned for this task. In [5], the authors proposed an inverse multi-path fingerprinting approach for beam-alignment utilizing prior measurements at a *given position* to provide a set of candidate beam directions at the *same position*.

However, by requiring measurements to already be available at a certain position for predictions to be made, this approach fails to predict the channel in those positions where measurements are not yet available. For this reason, this scheme requires to collect a huge amount of channel propagation measurements to cover the entire operational region, which may not be practical. In many practical settings, the spatial correlation in the channel may be exploited to provide beam directions recommendations also in *new* positions, where prior measurements are unavailable. To address this more general problem, in this work we leverage the tensor completion technique. This problem has been recently investigated in many areas, such as computer vision, image in-painting, recommendation systems, etc. [10].

By exploiting the low-rank of mmWave MIMO channels [1], [2], [9], we construct a data model on a subset of positions as a tensor and formulate the tensor completion problem to estimate the channel on those positions and beam directions where measurements are missing. To capture the channel spatial correlation, we introduce a smooth constraint that induces similarity among adjacent positions and beams. We propose a two-stage tensor completion algorithm composed of two smooth matrix completions [10] and a greedy selection algorithm to recommend a subset of candidate beams. We show numerically that our proposed beam recommendation algorithm can provide accurate beam candidates with small training overhead. Numerical evaluations demonstrate that our proposed method achieves 80% probability of correct alignment with only 2% of trained beams, given power measurements on only 20% of discrete random users' positions, as opposed to the state-of-the-art inverse-fingerprinting algorithm [5], which attains a probability of correct alignment of 50%.

The rest of this paper is organized as follows. In Sec. II, we present the system model and architecture; in Sec. III, we propose the recommendation algorithm with two-stage tensor completion. The numerical result are presented in Sec. IV, followed by concluding remarks in Sec. V.

II. SYSTEM MODEL

In this section, we describe the channel model and the beamforming codebook of our communication system. Then, we introduce the position-aided beam-alignment protocol and explain the goal of this work. Later, we depict the data collection and the data tensor.

The authors are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA; emails: {chou59, michelus, djlove, jvk}@purdue.edu.

This research has been funded by NSF under grant CNS-1642982.

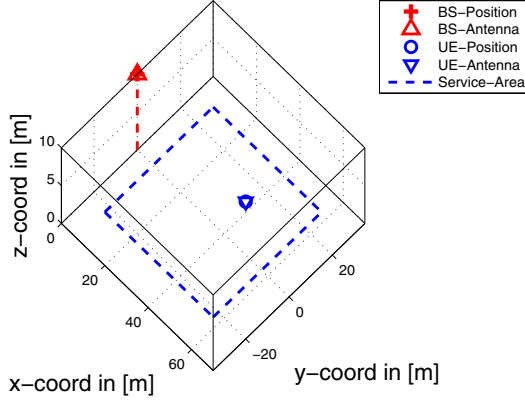


Fig. 1: The network layout of the scenario.

A. Channel Model

We consider a scenario with a base station (BS), servicing an area with GPS coordinates $\{(g_x, g_y) : X_0 \leq g_x \leq X_{end}, Y_0 \leq g_y \leq Y_{end}\}$, as in Fig. 1. The geometric channel model [2] is assumed for the uplink SIMO channel between the BS and the user (UE) at GPS coordinate $\mathbf{g} = (g_x, g_y)$ and is given by

$$\mathbf{h}^{\mathbf{g}} = \sqrt{N_r} \sum_{\ell=1}^L \alpha_{\ell}^{\mathbf{g}} \mathbf{a}_r(\theta_{\ell}^{\mathbf{g}}, \phi_{\ell}^{\mathbf{g}}),$$

where $\mathbf{a}_r(\theta_{\ell}^{\mathbf{g}}, \phi_{\ell}^{\mathbf{g}})$ (see (1)) is the normalized receive steering vector of the ℓ th path; $\theta_{\ell}^{\mathbf{g}}$ and $\phi_{\ell}^{\mathbf{g}}$ are its elevation and azimuth angles; $\alpha_{\ell}^{\mathbf{g}}$ is the complex channel gain; L is the number of paths; and N_r is the number of receive antennas.

In order to develop our data-driven approach, we aim to leverage the channel correlation with respect to some features of the environment where the UE is operating. Here, we consider the correlation between the channel and the UE position. To the best of our knowledge, the modeling of the propagation channel can only be achieved by real channel measurements or by simulation in ray-tracing software, which requires accurate modeling of the propagation environment, such as position of buildings, scatterers, etc. In practice, our algorithm is applicable to a fully data-driven approach based on actual channel measurements. For evaluation purposes, in the numerical results in Sec. IV, we will generate these measurements with Quadriga [11].

B. Beam Codebook and Received Signal Model

We consider a uniform planar array (UPA) [2], [5] at the BS with N_x and N_y antennas and $\lambda/2$ antenna spacing along the x and y directions (a total of $N_r = N_x N_y$ antennas), and a receive beamforming codebook

$$\mathcal{W} = \{\mathbf{w}_{i,j} = \mathbf{a}(\theta_i, \phi_j), i = 1, \dots, C_{\theta}, j = 1, \dots, C_{\phi}\},$$

of size $|\mathcal{W}| = C_{\theta} C_{\phi}$; $\mathbf{a}(\theta, \phi)$ is the array response vector representing a beam pointing in the elevation angle $\theta \in [-\pi/2, \pi/2)$ and the azimuth angle $\phi \in [-\pi/2, \pi/2)$,

$$\mathbf{a}(\theta, \phi) = \frac{1}{\sqrt{N_r}} \begin{bmatrix} 1 & e^{j\Omega_y} & \dots & e^{j(N_y-1)\Omega_y} \end{bmatrix}^T \otimes \begin{bmatrix} 1 & e^{j\Omega_x} & \dots & e^{j(N_x-1)\Omega_x} \end{bmatrix}^T, \quad (1)$$

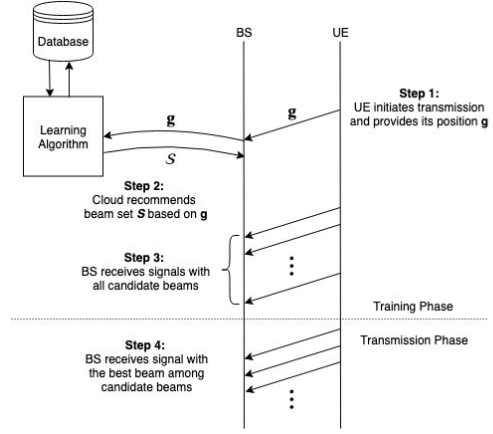


Fig. 2: Position-aided beam alignment protocol.

with $\Omega_y = \pi \sin \theta \sin \phi$, $\Omega_x = \pi \sin \theta \cos \phi$. To construct \mathcal{W} , θ_i and ϕ_j are uniformly quantized in $[-\pi/2, \pi/2)$ with resolution π/C_{θ} and π/C_{ϕ} as

$$\theta_i = -\frac{\pi}{2} + (i-1) \times \frac{\pi}{C_{\theta}}, \quad i = 1, \dots, C_{\theta}, \quad (2)$$

$$\phi_j = -\frac{\pi}{2} + (j-1) \times \frac{\pi}{C_{\phi}}, \quad j = 1, \dots, C_{\phi}. \quad (3)$$

We index the beamforming vectors in \mathcal{W} as

$$\mathcal{I} \equiv \{(i, j) : i = 1, \dots, C_{\theta}, j = 1, \dots, C_{\phi}\}.$$

We consider an uplink beam training scheme, in which the UE at GPS coordinate $\mathbf{g} = (g_x, g_y)$ transmits a known unit-norm training sequence vector $\mathbf{s} \in \mathbb{C}^{N \times 1}$ and the BS processes the received signal with the beamforming vector $\mathbf{w}_{i,j}$, yielding the received signal vector

$$\mathbf{y}_{i,j}^{\mathbf{g}} = \sqrt{P_t} \mathbf{w}_{i,j}^H \mathbf{h}^{\mathbf{g}} \mathbf{s} + \mathbf{v},$$

where $\mathbf{v} \sim \mathcal{CN}(0, \sigma_v^2 \mathbf{I})$ is the received noise vector; P_t is the transmit power. The received power with the (i, j) -th beamformer can be estimated as

$$r_{i,j}^{\mathbf{g}} = |\mathbf{s}^H \mathbf{y}_{i,j}^{\mathbf{g}}|^2 = |\sqrt{P_t} \mathbf{w}_{i,j}^H \mathbf{h}^{\mathbf{g}} + \tilde{v}|^2,$$

where $\tilde{v} = \mathbf{s}^H \mathbf{v}$ is zero-mean complex Gaussian noise with variance σ_v^2 . These received powers are stored in a database explained in Sec. II-D, and then used in our completion framework to predict the received power in other positions and beams.

C. Position-Aided Beam-Alignment

The idea of this approach is to provide a set of candidate beams at a given UE position. Since the overhead of the conventional beam-sweeping approach is unacceptable (it scales with $|\mathcal{W}|$ and is typically very large), our objective is to design a learning algorithm that provides a small subset $\mathcal{S} \subset \mathcal{W}$ of candidate beams for training, which is likely to contain the best beam (the one with highest received power).

In Fig. 2, we introduce a flow diagram for the position-aided beam-alignment. In step 1, the UE initiates the uplink mmWave transmission request accompanied with its GPS coordinate $\mathbf{g} = (g_x, g_y)$ to the BS using sub-6GHz control channels. The position information is available via a suite of sensors such as GPS or LIDAR [5], [12]. In step 2,

the BS forwards the UE's GPS coordinate \mathbf{g} to the cloud, which then processes the learning algorithm and provides the recommended beam set \mathcal{S} . In step 3, the UE transmits a sequence of $|\mathcal{S}|$ known signals, and the BS receives these signals with the codewords in the recommended beam set \mathcal{S} . Then, the BS selects the best beam (ranked by received power) among the recommended beams. In step 4, the BS uses the selected receive beamforming vector for the subsequent mmWave uplink data transmission. Due to the reciprocity of the wireless channel, the BS can also utilize these beam directions for downlink transmissions.

D. Data Model

Data is an essential element for the machine learning approach. Here, we describe how we store information in the database. We discretize the service area of the BS with resolution Δ_s and define the position labels $\mathbf{p} = (p_x, p_y)$ as the function of GPS coordinate \mathbf{g} ,

$$\mathbf{p}(\mathbf{g}) = \left(1 + \left\lfloor \frac{g_x - X_0}{\Delta_s} \right\rfloor, 1 + \left\lfloor \frac{g_y - Y_0}{\Delta_s} \right\rfloor \right), \quad (4)$$

where $\lfloor x \rfloor$ denotes the nearest integer to x ; the x-axis label $p_x \in \{1, \dots, L_x\}$ where $L_x = \left\lceil \frac{X_{\text{end}} - X_0}{\Delta_s} \right\rceil$; and the y-axis label $p_y \in \{1, \dots, L_y\}$ where $L_y = \left\lceil \frac{Y_{\text{end}} - Y_0}{\Delta_s} \right\rceil$. The BS measures the received power based on a fixed transmit power P_t . We define the received power with beam $\mathbf{w}_{i,j}$ at position \mathbf{p} as $r(\mathbf{p}(\mathbf{g}), i, j) = r_{i,j}^{\mathbf{g}}$. During the data collection, the BS might collect multiple measurements for the same beamforming vector and position. For beam $\mathbf{w}_{i,j}$ and position \mathbf{p} , we define $r_k^{(\mathbf{p}, i, j)}$ as the k -th measured received power and $N_{ob}^{(\mathbf{p}, i, j)}$ as the number of measurements collected so far on that position and beam index. We extract the average power information by computing the sample mean $\bar{r}^{(\mathbf{p}, i, j)} = \frac{1}{N_{ob}^{(\mathbf{p}, i, j)}} \sum_{k=1}^{N_{ob}^{(\mathbf{p}, i, j)}} r_k^{(\mathbf{p}, i, j)}$. Once the BS performs a new measurement for position \mathbf{p} and beam $\mathbf{w}_{i,j}$, we can update the average received power in an online fashion as

$$\bar{r}^{(\mathbf{p}, i, j)} \leftarrow \frac{N-1}{N} \bar{r}^{(\mathbf{p}, i, j)} + \frac{1}{N} r_N^{(\mathbf{p}, i, j)},$$

where $N = N_{ob}^{(\mathbf{p}, i, j)} + 1$, followed by $N_{ob}^{(\mathbf{p}, i, j)} \leftarrow N_{ob}^{(\mathbf{p}, i, j)} + 1$. Then, the database records the average received power along with the side information, including the UE's position $\mathbf{p} = (p_x, p_y)$, and the indices of the beamforming codeword $\mathbf{w}_{i,j}$, as in TABLE I.

TABLE I: Database form

p_x	p_y	i	j	$\bar{r}^{(\mathbf{p}, i, j)}$
1	1	1	4	5.2
1	2	4	5	6.1
\vdots	\vdots	\vdots	\vdots	\vdots

We represent the extracted data as a 4-th order tensor

$$\mathcal{T}(p_x, p_y, i, j) = \begin{cases} \bar{r}^{(p_x, p_y, i, j)}, & (p_x, p_y, i, j) \in \Psi, \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where Ψ is the set of observed combinations of positions and beams stored in the database and the unobserved entries $(p_x, p_y, i, j) \notin \Psi$ are set to zero. It is impractical to collect

the information with all combinations of positions and beam-directions into the database due to the limited sampling resources. For this reason, some positions possibly have no representation in the database. Even in the observed positions, there might be only a limited number of beams' information recorded. Therefore, the tensor \mathcal{T} may be highly incomplete.

III. TENSOR COMPLETION AND BEAM RECOMMENDATION

Our goal is to recommend a set \mathcal{S} of N_{tr} candidate beams for the UE to train based on its position. If the UE is in a position \mathbf{p} represented in the database, and with all beam measurements available, $\bar{r}^{(\mathbf{p}, p_y, i, j)}, \forall (i, j)$, then this task can be easily accomplished by recommending the N_{tr} beams with highest average received power in the given position.

Otherwise, we design an algorithm based on tensor completion that employs the knowledge at neighboring positions to support the beam recommendation for the UE. Note that successful completion is highly dependent on the sampling set. If measurements are missing on a certain row or column of an incomplete matrix, then no reconstruction is possible on that row or column, if we only rely on a low-rank approximation [10]. A similar issue also exists in the tensor case: if no measurements are available on a certain index in a given dimension, no elements corresponding to this index can be predicted by only relying on low rank structure. In our problem, the database contains measurements related to a subset of beams on few observed positions. The data tensor might be so incomplete that we cannot guarantee that every index of each dimension is measured at least once. Therefore, the low-rank tensor completion often fails to provide predictions on unobserved positions.

To address this challenge, in addition to the low-rank approximation, we enforce a smoothness constraints across adjacent entries on a given dimension. This constraint captures realistic spatial correlations across adjacent positions arising in mmWave channels: similarity between neighboring beams at a given position; and similarity between neighboring positions on a given beam direction. In Sec. III-A, we propose a two-stage tensor completion implemented by dividing the tensor completion into two smooth matrix completions (SMCs), considered in Sec. III-B. Finally, we propose a greedy algorithm to provide the recommended beams based on the predicted received power in Sec. III-C.

A. Two-stage Tensor Completion

Given the data tensor \mathcal{T} in (5) and the set of observed combinations of positions and beams Ψ , we aim to recover the incomplete tensor \mathcal{T} . Since the tensor \mathcal{T} is highly incomplete, we might only have limited number of beams' information on few observed positions, causing the low-rank completion to fail. To address this challenge, we propose a two-stage tensor completion, each based on SMC.

In the first stage, for each observed position (x_o, y_o) such that $(x_o, y_o, i, j) \in \Psi$ for some (i, j) , we do the SMC on the beam matrix to predict the received power on the unobserved beams, by exploiting the low-rank property that the received powers of beams in a given position tend to concentrate in few beam clusters due to the limited scattering of mmWave channels. The smoothness between neighboring beams depends on the beamwidth of the receive beamforming and the angular spread of the channel [13]. To this end,

let $\mathbf{B}^{(x_o, y_o)} = \mathcal{T}(x_o, y_o, :, :)$ be the (possibly incomplete) matrix of received powers along beam directions, for the given position (x_o, y_o) ; let $\Omega \equiv \{(i, j) : (x_o, y_o, i, j) \in \Psi\}$ be the set of observed beams in position (x_o, y_o) . Then, the SMC can be expressed as

$$\text{SMC}_\Omega(\mathbf{B}^{(x_o, y_o)}) = \arg \min_{\mathbf{X}} \|\mathbf{X}\|_* + \gamma(\|\mathbf{D}_{C_\theta} \mathbf{X}\|_F^2 + \|\mathbf{X} \mathbf{D}_{C_\phi}^T\|_F^2) \quad (6)$$

s.t. $\mathbf{X}_\Omega = \mathbf{B}_\Omega^{(x_o, y_o)}$

and is considered in Sec. III-B. The first term of the objective function is the nuclear norm of \mathbf{X} , $\|\mathbf{X}\|_* = \sum_{i=1}^{\min(m, n)} \sigma_i$, where σ_i is the i -th largest singular value of \mathbf{X} . The second term of the objective function is a penalty term which induces smoothness across entries in each row and column of \mathbf{X} . The matrix $\mathbf{D}_m \in \mathbb{R}^{(m-1) \times m}$ is the smoothness matrix, capturing the differences between neighboring entries of a matrix:

$$\mathbf{D}_m = \begin{bmatrix} 1 & -1 & \cdots & 0 & 0 \\ \vdots & 1 & -1 & \vdots & 0 \\ 0 & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & -1 \end{bmatrix}_{(m-1) \times m}. \quad (7)$$

Thus, $\|\mathbf{X} \mathbf{D}_n^T\|_F^2$ and $\|\mathbf{D}_m \mathbf{X}\|_F^2$ quantify the row and column smoothness of matrix \mathbf{X} , respectively. In our optimization, we consider smoothness on rows and columns simultaneously, as opposed to LTVNN [14], which considers them separately.

After the first stage completion by SMC, we obtain the data tensor \mathcal{T}' completed on the observed positions and update Ψ' by setting all predicted terms as observed. In the second stage, the SMC is implemented on the position matrix for each beam, by leveraging the fact that the received power tends to vary smoothly between neighboring positions on a given beam. For a given beam, the smoothness between neighboring positions is related to the position resolution Δ_s . Specifically, in each beam (i_o, j_o) such that $(p_x, p_y, i_o, j_o) \in \Psi'$ for some (p_x, p_y) , let $\mathbf{G}^{(i_o, j_o)} = \mathcal{T}'(:, :, i_o, j_o)$ be the (possibly incomplete) matrix of received powers on different positions, along the given beam indexed by (i_o, j_o) ; let $\Omega \equiv \{(p_x, p_y) : (p_x, p_y, i_o, j_o) \in \Psi'\}$ be the set of observed positions along the beam (i_o, j_o) . Then, the SMC can be expressed as

$$\text{SMC}_\Omega(\mathbf{G}^{(i_o, j_o)}) = \arg \min_{\mathbf{X}} \|\mathbf{X}\|_* + \gamma(\|\mathbf{D}_{L_x} \mathbf{X}\|_F^2 + \|\mathbf{X} \mathbf{D}_{L_y}^T\|_F^2) \quad (8)$$

s.t. $\mathbf{X}_\Omega = \mathbf{G}_\Omega^{(i_o, j_o)}$

and is considered in Sec. III-B. After the second stage completion by SMC, we get the completed tensor $\hat{\mathcal{T}}$, which predicts the received power of all unknown positions/beams, and is used in Sec. III-C to recommend the set of training beams at a given UE position \mathbf{p} . The two-stage tensor completion algorithm is shown in **Algorithm 1**. Next, we discuss the solutions of the SMC problems (6) and (8), solved by **Algorithm 2**.

B. Smooth Matrix Completion (SMC)

The smooth matrix completion exploits both the low rank and the smoothness of the data. Given the incomplete matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ with $\mathbf{M}_{ij}, \forall (i, j) \in \Omega$, the SMC problem to predict the unobserved entries $\mathbf{M}_{ij}, \forall (i, j) \notin \Omega$ is expressed as

$$\min_{\mathbf{X}} \|\mathbf{X}\|_* + \gamma(\|\mathbf{D}_m \mathbf{X}\|_F^2 + \|\mathbf{X} \mathbf{D}_n^T\|_F^2) \quad \text{s.t. } \mathbf{X}_\Omega = \mathbf{M}_\Omega, \quad (9)$$

where γ is a regularization parameter. We use the alternating direction method of multipliers (ADMM) [15] to efficiently solve (9). With ADMM, we reformulate the problem as

$$\min_{\mathbf{X}, \mathbf{Y}} \|\mathbf{X}\|_* + \gamma(\|\mathbf{D}_m \mathbf{Y}\|_F^2 + \|\mathbf{Y} \mathbf{D}_n^T\|_F^2) + \frac{\lambda}{2} \|\mathbf{Y} - \mathbf{X}\|_F^2 \quad (10)$$

s.t. $\mathbf{Y}_\Omega = \mathbf{M}_\Omega, \quad \mathbf{X} = \mathbf{Y},$

where $\lambda > 0$ is a small fixed parameter. We introduce the Lagrangian multiplier \mathbf{Z} associated with the constraint $\mathbf{X} = \mathbf{Y}$. The augmented Lagrange function of (10) is

$$L(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \|\mathbf{X}\|_* + \gamma(\|\mathbf{D}_m \mathbf{Y}\|_F^2 + \|\mathbf{Y} \mathbf{D}_n^T\|_F^2) + \text{tr}(\mathbf{Z}^T (\mathbf{Y} - \mathbf{X})) + \frac{\lambda}{2} \|\mathbf{Y} - \mathbf{X}\|_F^2. \quad (11)$$

The ADMM algorithm is implemented by minimizing iteratively $L(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ over \mathbf{X} and \mathbf{Y} , and then update \mathbf{Z} as

$$\begin{aligned} \mathbf{X}_{t+1} &= \arg \min_{\mathbf{X}} L(\mathbf{X}, \mathbf{Y}_t, \mathbf{Z}_t); \\ \mathbf{Y}_{t+1} &= \arg \min_{\mathbf{Y}} L(\mathbf{X}_{t+1}, \mathbf{Y}, \mathbf{Z}_t), \quad \text{s.t. } \mathbf{Y}_\Omega = \mathbf{M}_\Omega; \\ \mathbf{Z}_{t+1} &= \mathbf{Z}_t + \beta(\mathbf{Y}_{t+1} - \mathbf{X}_{t+1}); \end{aligned} \quad (12)$$

where β is a step-size. To optimize \mathbf{X} , we minimize $L(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ with fixed \mathbf{Y}_t and \mathbf{Z}_t , yielding

$$\mathbf{X}_{t+1} = \arg \min_{\mathbf{X}} \|\mathbf{X}\|_* + \frac{\lambda}{2} \left\| \mathbf{X} - \left(\mathbf{Y}_t + \frac{1}{\lambda} \mathbf{Z}_t \right) \right\|_F^2.$$

In [10], it is shown that this problem is strictly convex and its solution is given by the singular value thresholding, $\mathbf{X}_{t+1} = \mathcal{D}_{1/\lambda}(\mathbf{Y}_t + \frac{1}{\lambda} \mathbf{Z}_t)$, where \mathcal{D}_τ is the soft-thresholding operator. For a matrix \mathbf{A} with singular value decomposition (SVD) $\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^H$, where $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_r)$, this is defined as $\mathcal{D}_\tau(\mathbf{A}) = \mathbf{U} \mathcal{D}_\tau(\mathbf{\Sigma}) \mathbf{V}^H$, $\mathcal{D}_\tau(\mathbf{\Sigma}) = \text{diag}(\{\max\{\sigma_i - \tau, 0\}\})$.

The minimization of $L(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ over \mathbf{Y} with fixed \mathbf{X}_{t+1} and \mathbf{Z}_t can be formulated as

$$\begin{aligned} \mathbf{Y}_{t+1} &= \arg \min_{\mathbf{Y}} \gamma(\|\mathbf{D}_m \mathbf{Y}\|_F^2 + \|\mathbf{Y} \mathbf{D}_n^T\|_F^2) \\ &\quad + \text{tr}(\mathbf{Z}_t^T (\mathbf{Y} - \mathbf{X}_{t+1})) + \frac{\lambda}{2} \|\mathbf{Y} - \mathbf{X}_{t+1}\|_F^2 \quad (13) \\ \text{s.t. } \mathbf{Y}_\Omega &= \mathbf{M}_\Omega. \end{aligned}$$

To solve this problem, we restrict its optimization to the unobserved set $\bar{\Omega} \equiv \{(i, j) : (i, j) \notin \Omega\}$, and force $\mathbf{Y}_{ij} = \mathbf{M}_{ij}$ for $(i, j) \in \Omega$. By computing the derivative of (13) with respect to \mathbf{Y}_{ij} for $(i, j) \in \bar{\Omega}$ and setting it to zero, we obtain the equation $\text{tr}(\{\mathbf{U}^{(i, j)}\}^T \mathbf{Y}) - \frac{\lambda \mathbf{X}_{ij} - \mathbf{Z}_{ij}}{2\gamma} = 0$, where

$$\begin{aligned} \mathbf{U}^{(i, j)} &= \mathbf{D}_m^T \mathbf{D}_m \mathbf{e}_m(i) \mathbf{e}_n(j)^T + \mathbf{e}_m(i) \mathbf{e}_n(j)^T \mathbf{D}_n^T \mathbf{D}_n \\ &\quad + \frac{\lambda}{2\gamma} \mathbf{e}_m(i) \mathbf{e}_n(j)^T. \end{aligned}$$

The vector $\mathbf{e}_m(i)$ is an $m \times 1$ vector with the i -th element equal to 1 and all other elements equal to 0. We force $\mathbf{Y}_\Omega = \mathbf{M}_\Omega$ as in (13), yielding the set of $|\bar{\Omega}|$ linear equations, $\forall (i, j) \in \bar{\Omega}$,

$$\sum_{(p, q) \in \bar{\Omega}} \mathbf{U}_{pq}^{(i, j)} \mathbf{Y}_{pq} = \frac{\lambda \mathbf{X}_{ij} - \mathbf{Z}_{ij}}{2\gamma} - \sum_{(p, q) \in \Omega} \mathbf{U}_{pq}^{(i, j)} \mathbf{M}_{pq}. \quad (14)$$

There are $|\bar{\Omega}|$ unknowns $(\mathbf{Y}_{pq}, (p, q) \in \bar{\Omega})$ and $|\bar{\Omega}|$ linear equations, so the matrix $\mathbf{Y}_{\bar{\Omega}}$ can be derived by solving (14). Then, we update the Lagrangian multiplier \mathbf{Z} with fixed \mathbf{X}_{t+1} and \mathbf{Y}_{t+1} as in (12). The algorithm updates \mathbf{X} , \mathbf{Y} , and \mathbf{Z} iteratively.

actively until a stop criterion is satisfied. With the convergence of ADMM [16], the iteration approaches the feasibility $\|\mathbf{X}_t - \mathbf{Y}_t\|_F \rightarrow 0$, thus we set the stop condition as $\|\mathbf{X}_t - \mathbf{Y}_t\|_F \leq \epsilon$. It follows that $\|\mathbf{Z}_{t+1} - \mathbf{Z}_t\|_F = \beta \|\mathbf{Y}_{t+1} - \mathbf{X}_{t+1}\|_F \rightarrow 0$, which guarantees the convergence of \mathbf{Z} . The computational complexity of SMC is dominated by the SVD to perform the soft-thresholding operation in each iteration, which is $\mathcal{O}(mn^2)$ for $m \geq n$ [17]. The update of \mathbf{Y} requires a matrix inversion per iteration, whose computational complexity is $\mathcal{O}(|\bar{\Omega}|^3)$.

Algorithm 1 Two-stage Tensor Completion

Input: incomplete data tensor \mathcal{T} and the observed set Ψ

Output: $\hat{\mathcal{T}}$

```

1: for  $(x_o, y_o, -, -) \in \Psi$  do
2:    $\mathbf{B}^{(x_o, y_o)} = \mathcal{T}(x_o, y_o, :, :)$ 
3:    $\Omega = \{(i, j) | (x_o, y_o, i, j) \in \Psi\}$ 
4:    $\mathcal{T}'(x_o, y_o, :, :) = \text{SMC}_{\Omega}(\mathbf{B}^{(x_o, y_o)})$ 
5:   Update the observed set  $\Psi'$ 
6: end for
7: for  $(-, -, i_o, j_o) \in \Psi'$  do
8:    $\mathbf{G}^{(i_o, j_o)} = \mathcal{T}'(:, :, i_o, j_o)$ 
9:    $\Omega = \{(p_x, p_y) | (p_x, p_y, i_o, j_o) \in \Psi'\}$ 
10:   $\hat{\mathcal{T}}(:, :, i_o, j_o) = \text{SMC}_{\Omega}(\mathbf{G}^{(i_o, j_o)})$ 
11: end for

```

Algorithm 2 Smooth Matrix Completion (SMC)

Input: incomplete data matrix \mathbf{M} and observed set Ω

Output: $\hat{\mathbf{M}}$

```

1: Initialization  $\mathbf{X}_t = \mathbf{Y}_t = \mathbf{M}$ ,  $\mathbf{Z}_t = \mathbf{0}$ ,  $\epsilon_t = \infty$ 
2: while  $\epsilon_t > \epsilon$  do
3:    $\mathbf{X}_{t+1} = \mathcal{D}_{1/\lambda}(\mathbf{Y}_t + \frac{1}{\lambda} \mathbf{Z}_t)$ 
4:    $\mathbf{Y}_{t+1} = \mathbf{Y}_{\bar{\Omega}} + \mathbf{M}_{\bar{\Omega}}$  ( $\mathbf{Y}_{\bar{\Omega}}$  obtained by solving (14))
5:    $\mathbf{Z}_{t+1} = \mathbf{Z}_t + \beta(\mathbf{Y}_{t+1} - \mathbf{X}_{t+1})$ 
6:    $\epsilon_{t+1} = \|\mathbf{X}_{t+1} - \mathbf{Y}_{t+1}\|_F$ ;  $t := t + 1$ 
7: end while
8:  $\hat{\mathbf{M}} \leftarrow \mathbf{X}_{\bar{\Omega}} + \mathbf{M}_{\bar{\Omega}}$ 

```

C. Recommendation Algorithm

With the completed tensor $\hat{\mathcal{T}}$, we have the estimated received powers of all beams at UE position (p_x, p_y) . Suppose the number of beams to be trained is N_{tr} , the construction of the recommended beam set is a beam subset selection problem. It can be directly fulfilled by selecting the N_{tr} beams with largest predicted received power from the completed tensor $\hat{\mathcal{T}}$ using **Algorithm 3**.

Algorithm 3 Beam Subset Selection

Input: completed tensor $\hat{\mathcal{T}}$, beam number N_{tr} , beam codebook \mathcal{W} with indices \mathcal{I} , UE position (p_x, p_y)

Output: recommended beam subset $\mathcal{S}_{N_{tr}}$

```

1: Initialization  $\mathcal{S}_0 \leftarrow \emptyset$ 
2: for  $n = 1 : N_{tr}$  do
3:    $(i^*, j^*) = \arg \max_{(i, j) \in \mathcal{I} \setminus \mathcal{S}_{n-1}} \hat{\mathcal{T}}(p_x, p_y, i, j)$ 
4:    $\mathcal{S}_n \leftarrow \mathcal{S}_{n-1} \cup (i^*, j^*)$ 
5: end for

```

IV. NUMERICAL RESULTS

Here, we evaluate the misalignment probability and the spectral efficiency of our proposed beam recommendation algorithm (TC) with the channel generated by Quadriga [11].

A. Experiment Setting

We consider an uplink SIMO scenario, with an UPA having $N_x = N_y = 16$ antennas along the x and y directions (as in (1)) at the BS, and isotropic antenna at the UE. The scenario *mmMAGIC_UMi_NLOS* is selected, with carrier frequency $f_c = 58.68$ GHz. The UPA codebook size is $|\mathcal{W}| = 256$ with $(C_{\theta}, C_{\phi}) = (16, 16)$. The network layout is depicted in Fig. 1, containing one BS at $(0, 0, 10)$ serving the UE in the area $\mathcal{A} = \{(g_x, g_y) : 10 \leq g_x \leq 60, -25 \leq g_y \leq 25\}$ with height as 1.5 m. Considering $51 \times 51 = 1261$ reference GPS coordinates uniformly located in the service area \mathcal{A} , we collect the SIMO channel for each reference GPS coordinate in \mathcal{A} as the ground truth data. The position labels of \mathcal{A} are derived as in (4) with the resolution $\Delta_s = 5\text{m}$, where the length are $L_x = 11$ and $L_y = 11$. The data tensor can be expressed by $\mathcal{T} \in \mathbb{R}^{L_x \times L_y \times C_{\theta} \times C_{\phi}}$, where $(L_x, L_y, C_{\theta}, C_{\phi}) = (11, 11, 16, 16)$. The observed position ratio $K_{op} = C_{op}/(L_x L_y)$ is varied, where C_{op} denotes the number of observed positions. Regarding the observed set of data tensor, we make the two following assumptions for the experiment setting. **Assumption 1:** The observed positions are randomly chosen. For the observed position \mathbf{p}' , the measurements of the reference GPS coordinates corresponding to position \mathbf{p}' , $\{\mathbf{g} = (g_x, g_y) : \mathbf{p}(\mathbf{g}) = \mathbf{p}', \mathbf{g} \in \mathcal{A}\}$, are observed. **Assumption 2:** For each observed GPS coordinate \mathbf{g} , only the measurements of the top 10% beams (ranked by received power) are stored in the database. With these two assumptions, the data tensor in (5) is incomplete in both positions' and beams' dimensions.

Our formulation allows to make predictions for unknown beams/positions by exploiting spatial correlation. On the other hand, the previous work [5] for mmWave beam alignment uses the prior knowledge already available at a given position, but does not allow to make predictions if measurements are not available. For comparison, we consider the type B fingerprinting method [5] by providing the recommended beam set based on the closest position having available prior knowledge, if the prior measurements of UE position are not given.

B. Performance of Proposed Beam-alignment Algorithm

We evaluate the power loss probability $P_{pl}(\mathcal{S}_{\mathbf{p}})$ versus the percentage of trained beams ($|\mathcal{S}_{\mathbf{p}}|/|\mathcal{W}|$) under different observed position ratio K_{op} . The noise impact is ignored. To measure the beam alignment accuracy for the recommended set $\mathcal{S}_{\mathbf{p}}$, we define the metric as the power loss probability $P_{pl}(\mathcal{S}_{\mathbf{p}}) = 1 - P_s(\mathcal{S}_{\mathbf{p}})$, where $P_s(\mathcal{S}_{\mathbf{p}})$ is the probability that the best beam is included in the set $\mathcal{S}_{\mathbf{p}}$ at position \mathbf{p} ,

$$P_s(\mathcal{S}_{\mathbf{p}}) = \mathbb{P} \left(\max_{(i, j) \in \mathcal{S}_{\mathbf{p}}} r^{(\mathbf{p}, i, j)} = \max_{(i', j') \in \mathcal{I}} r^{(\mathbf{p}, i', j')} \right). \quad (15)$$

We average the power loss probability over the channels at the GPS coordinates corresponding to unobserved positions. In Fig. 3, $P_{pl}(\mathcal{S}_{\mathbf{p}})$ decreases when the database contains more observed positions with fixed number of trained beams. The type-B fingerprinting method [5] requires at least 13% of the beams to be trained to attain $P_{pl}(\mathcal{S}_{\mathbf{p}}) = 20\%$. With K_{op} increasing

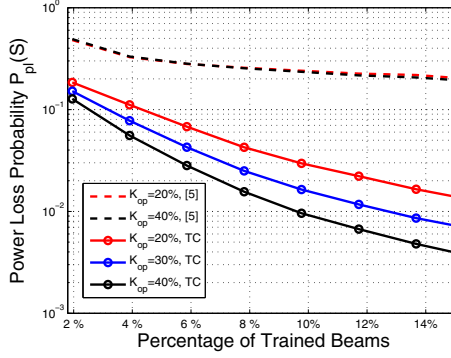


Fig. 3: Power loss probability versus the percentage of trained beams.

from 20% to 40%, $P_{pl}(S_p)$ only very slightly improves. For our proposed method (TC), to attain $P_{pl}(S_p) = 20\%$, we require only 2% trained beams when $K_{op} = 20\%$. With more trained beams or higher K_{op} , $P_{pl}(S_p)$ of TC decreases significantly.

C. Spectral Efficiency

We evaluate the average spectral efficiency versus the average transmit power. We first define the transmission rate as

$$R = B \log_2(1 + \eta P_t \|\mathbf{w}^H \mathbf{h}\|^2) \quad (16)$$

where $B = 1.76$ GHz is the bandwidth [5], [11]; P_t is the transmit power; \mathbf{w} is the selected beamforming vector; \mathbf{h} is the SIMO channel; $\eta \triangleq \frac{\Lambda^2 \zeta}{8\pi d^2 N_0 B}$ is the SNR scaling factor, where $\Lambda = c/f_c$ is the wavelength (c is the speed of light), $N_0 = -174$ dBm/Hz is the noise power spectral density, d is the distance, and $\zeta = 1$ is the antenna efficiency. The microslot duration $\delta_S = 10 \mu s$ is the time for one transmission. The frame time $T_{frame} = 5$ ms is fixed. The training time is $T_{train} = N_{tr} \times \delta_S$. The fraction of time used for data transmission is $f_{comm} = \frac{T_{frame} - T_{train}}{T_{frame}}$. Then, the average throughput is $\bar{R} = R \times f_{comm}$.

In Fig. 4, the trend of spectral efficiency (\bar{R}/B) is monotone increasing. We compare our proposed scheme (TC) with the type-B fingerprinting method [5] and the *exhaustive search* which trains all $|\mathcal{W}|$ beams in the training phase. TC is better than *exhaustive search*. The spectral efficiency of TC with $(K_{op}, N_{tr}) = (40\%, 10)$ is around twice as much as the spectral efficiency of *exhaustive search*. It is due to $f_{comm} \approx 0.5$ for *exhaustive search*, but f_{comm} of TC is close to 1 because of the small T_{train} . The spectral efficiency of TC at $(K_{op}, N_{tr}) = (20\%, 5)$ outperforms the type B fingerprinting method at $(K_{op}, N_{tr}) = (40\%, 10)$ by 0.7 bit/s/Hz since our method provides more accurate beam prediction with even fewer known positions and fewer trained beams. If we increase K_{op} or N_{tr} , the improvement of TC is minor since $P_{pl}(S)$ is fairly low ($< 10\%$) in this region.

V. CONCLUSION

In this paper, we propose a learning-based beam recommendation algorithm to reduce the training overhead for the position-aided beam alignment protocol. We consider a scenario where the UE is located in an arbitrary position in which prior measurements may not be available in the database. We propose a two-stage tensor completion to predict

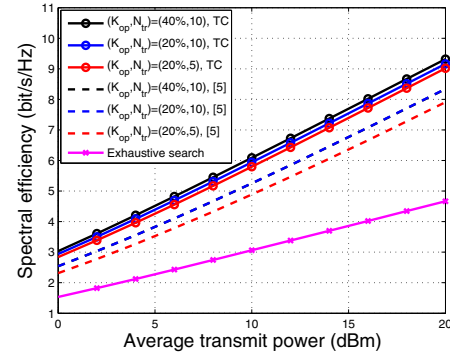


Fig. 4: Spectral efficiency (\bar{R}/B) versus the average transmit power (P_t).

the received power, and then provide the set of recommended beams by ranking the predicted power of beams. The two-stage tensor completion exploits both the low-rank and smoothness of the data. The numerical results demonstrate that the proposed beam recommendation algorithm does improve the performance of beam-alignment over the state-of-the-art, by reducing the beam-training overhead.

REFERENCES

- [1] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5g cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.
- [2] R. W. Heath *et al.*, "An overview of signal processing techniques for millimeter wave mimo systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 436–453, April 2016.
- [3] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive mimo for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, February 2014.
- [4] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5g," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, February 2014.
- [5] V. Va, J. Choi, T. Shimizu, G. Bansal, and R. W. Heath, "Inverse multipath fingerprinting for millimeter wave v2i beam alignment," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4042–4058, May 2018.
- [6] S. Hur, T. Kim, D. J. Love, J. V. Krogmeier, T. A. Thomas, and A. Ghosh, "Millimeter wave beamforming for wireless backhaul and access in small cell networks," *IEEE Trans. on Communications*, vol. 61, no. 10, pp. 4391–4403, October 2013.
- [7] M. R. Akdeniz *et al.*, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164–1179, June 2014.
- [8] M. Hussain and N. Michelusi, "Energy-efficient interactive beam alignment for millimeter-wave networks," *IEEE Trans. on Wireless Communications*, vol. 18, no. 2, pp. 838–851, Feb 2019.
- [9] A. Alkhateeb *et al.*, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct 2014.
- [10] J. feng Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optimization*, 2010.
- [11] S. Jaekel *et al.*, "Quadriga: A 3-d multi-cell channel model with time evolution for enabling virtual field trials," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 6, pp. 3242–3256, June 2014.
- [12] E. Ward and J. Folkesson, "Vehicle localization with low cost radar sensors," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 864–870.
- [13] J. Lee *et al.*, "Field-measurement-based received power analysis for directional beamforming millimeter-wave systems: Effects of beamwidth and beam misalignment," *ETRI Journal*, vol. 40, no. 1, pp. 26–38, 2018.
- [14] X. Han *et al.*, "Linear total variation approximate regularized nuclear norm optimization for matrix completion," *Abstract and Applied Analysis*, p. 8, 2014.
- [15] J. Yang and X. Yuan, "Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization," *Mathematics of Computation*, vol. 82, no. 281, pp. 301–329, 2013.
- [16] S. Boyd *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
- [17] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. The Johns Hopkins University Press, 2013.