Downlink Precoding for DP-UPA FDD
Massive MIMO via Multi-Dimensional Active
Channel Sparsification

Han Yu, Xinping Yi, and Giuseppe Caire

Abstract

In this paper, we consider user selection and downlink precoding for an over-loaded single-cell massive multiple-input multiple-output (MIMO) system in frequency division duplexing (FDD) mode, where the base station is equipped with a dual-polarized uniform planar array (DP-UPA) and serves a large number of single-antenna users. Due to the absence of uplink-downlink channel reciprocity and the high-dimensionality of channel matrices, it is extremely challenging to design downlink precoders using closed-loop channel probing and feedback with limited spectrum resource. To address these issues, a novel methodology – active channel sparsification (ACS) – has been proposed recently in the literature for uniform linear array (ULA) to design sparsifying precoders, which boosts spectral efficiency for multi-user downlink transmission with substantially reduced channel feedback overhead. Pushing forward this line of research, we aim to facilitate the potential deployment of ACS in practical FDD massive MIMO systems, by extending it from ULA to DP-UPA with explicit user selection and making the current ACS implementation simplified. To this end, by leveraging Toeplitz matrix theory, we start with the spectral properties of channel covariance matrices from the lens of their matrix-valued spectral density function. Inspired by these properties, we extend the original ACS using scale-weight bipartite graph representation to the matrix-weight counterpart. Building upon such matrix-weight bipartite graph representation, we propose a multi-dimensional ACS (MD-ACS) method, which is a generalization of original ACS formulation and is more suitable for DP-UPA antenna configurations. The nonlinear integer program formulation of MD-ACS can be classified as a generalized multi-assignment problem (GMAP), for which we propose a simple yet efficient greedy algorithm to solve it. Simulation results demonstrate

H. Yu and X. Yi are with Department of Electrical Engineering and Electronics, University of Liverpool, L69 3BX, United Kingdom. Email: {han.yu, xinping.yi}@liverpool.ac.uk.

G. Caire is with Communications and Information Theory Group (CommIT), Technical University of Berlin, 10587, Berlin, Germany. Email: {caire}@tu-berlin.de.
the performance improvement of the proposed MD-ACS with greedy algorithm over the state-of-the-art methods based on the QuaDRiGa channel models.

I. INTRODUCTION

Massive multiple-input multiple-output (MIMO) has been demonstrated, in both theory and practice, as one of the major performance boosters for the next generation (5G and beyond) wireless communication systems [1], [2]. Operating massive MIMO in time division duplex (TDD) mode is technically favorable, because the inherent uplink-downlink channel reciprocity makes it convenient to reconstruct the downlink channel vectors directly from the uplink pilot observations without requiring downlink training. Nevertheless, from the mobile network operators’ standpoint, the frequency division duplex (FDD) mode seems more preferable, as the current wireless systems are mainly operating in FDD mode, for which a lot of resource has been invested (e.g., in acquiring the spectrum), and FDD systems show a much better performance in scenarios with symmetric data traffic and delay-sensitive applications.

For FDD massive MIMO, however, the uplink-downlink channel reciprocity does not hold in general, due to the large uplink/downlink frequency separation that exceeds the fading coherence bandwidth. The base stations (BSs) have to probe the downlink channels via pilot training and ask for channel information feedback from the users. The high-dimensional channel vectors (due to the large number of antennas) incur prohibitively expensive feedback overhead and therefore result in inevitable performance degradation provided the limited channel coherence time and bandwidth. Recently, a fast-growing number of techniques have been proposed to make FDD as competitive as TDD systems by reducing downlink training and uplink feedback overhead, e.g., joint spatial division and multiplexing (JSDM) [3], [4]. It has been observed that the high-dimensional channel vectors admit a sparse representation in the angular/beam domain, such that they could be efficiently represented by low-dimensional ones [3]–[6]. As such, the pilot dimension of downlink training and the feedback overhead can be substantially reduced.

By exploiting the sparse representation in the beam domain, a number of techniques have been proposed. First, the compressed sensing (CS)-inspired methods (e.g., [5]–[7]) exploit this sparse representation to reconstruct the channel vectors at the BS using the compressed downlink pilot signals fed back from users. In particular, it allows each user to obtain the compressive measurements of the probing signals locally, and feed back to the BS, so that the BS can jointly recover the channel vectors using CS techniques [5]. Although effective to some extent, these CS-
based techniques rely highly on the knowledge of channel vector sparsity order (i.e., the number of significant elements), and probably fail to reconstruct downlink channels reliably when the devoted pilot dimension is less than the sparsity order. Second, channel reconstruction by exploiting the second-order statistics has attracted more and more attention [8]–[10]. Instead of feeding back compressed measurements of pilot signals, these approaches leverage the angular domain reciprocity to reconstruct downlink channel covariance matrices from the uplink training. Third, there is a new trend of using deep learning to predict downlink channel from the observations of uplink training [11]–[14]. The basic idea is to build up a mapping from uplink channel vectors to downlink ones by using an over-parameterized deep neural network. In principle, the deep neural networks with a sufficiently large number of parameters are able to approximate any complicated functions, as long as the training dataset is large enough. Nevertheless, there are still many challenges to design an efficient deep neural network for channel reconstruction.

More recently, channel reconstruction methodologies using second-order channel statistics have been advanced by e.g., [15]–[18], which aim to reconstruct downlink channel for FDD massive MIMO by exploiting the angular scattering function (ASF) reciprocity. This technique relies on the key assumption of the reciprocity of the ASF – it assumes that the ASF is frequency-invariant over both uplink and downlink frequency bands [10]. It consists of two major components: (1) Acquiring downlink channel support information in the angular domain from uplink channel training by exploiting uplink/downlink angular domain reciprocity; and (2) exploiting structural properties of such support information to design efficient downlink probing and uplink feedback schemes. As the channel angular support (i.e., non-zero elements) information is contained in the covariance matrix, its acquisition can be done by estimating downlink covariance matrices from the uplink ones, followed by the angular supports extraction. The channel support information of all users establishes a beam-user association (that can be modeled by a bipartite graph), in which the support of a user’s channel vector indicates the corresponding beams that can be utilized to serve this user. Such a beam-user association can be exploited for intelligent beam-user assignment that leads to artificially sparsified users’ channels. The active channel sparsification (ACS) will finally help reduce the pilot dimension for downlink probing, while allowing for simultaneous multiple-access of a large number of users using spatial multiplexing.

However, the ACS methodology is still facing some challenges in the potential deployment in the practical massive MIMO systems. On one hand, dual-polarized uniform planar array (DP-UPA) is commonly used in the practical systems, although attempts have been made to
extend from ULA to DP-ULA [16]. On the other hand, although the current ACS implementation using the mixed integer linear program (MILP) formulation is elegant in theory, its computational complexity scales as the number of antennas and users. To address these issues, in this paper, we consider to extend the ACS formulation from ULA/DP-ULA to DP-UPA by leveraging a matrix-weight bipartite graph representation for users’ channels. By relaxing the original MILP formulation, we come up with a new nonlinear integer program (NIP) formulation. Hence, we propose a greedy algorithm to solve the NIP problem in an efficient way. Specifically, our contributions are summarized as follows.

- The channel covariance matrix of massive MIMO with DP-UPA antennas can be recognized as a doubly block Toeplitz matrix. By leveraging Toeplitz matrix theory, we characterize the spectral properties of channel covariance matrices by investigating their matrix-valued spectral density function, which is also referred to as angular scattering function [19]. There exhibits some sparsity in the spectral density function when the angular spread is narrow under the context of DP-UPA massive MIMO scenarios.
- Inspired by these properties, we extend channel representation of ACS using bipartite graph from the original scale-weight to the matrix-weight counterpart. The matrix-weight bipartite graph establishes the association between block beams (correspond to dual-polarized antenna) and users according to the asymptotic block diagonalization of the channel covariance matrices. Building upon the matrix-weight bipartite graph representation, we propose a multi-dimensional ACS (MD-ACS) method, which is a generalized version of original ACS formulation and is more suitable for DP-UPA antenna configurations. The MD-ACS can be formulated as a generalized multi-assignment problem, which includes the original ACS formulation (i.e., assignment problem) as a special case.
- By taking into account the sum rate maximization and multiuser interference control, we reformulate the MD-ACS approach as a nonlinear integer program, for which we propose a simple yet efficient greedy algorithm to solve it. The extensive simulation results using QuaDRiGa channel models demonstrate the superiority of the proposed MD-ACS with greedy algorithm to the state-of-the-art methods, including the recently advanced ACS method concerning DP-ULA antenna configurations.

The rest of this paper is organized as follows. In the next section, we describe the channel and system model of the DP-UPA FDD massive MIMO system with downlink training and precoding.
In Section III, we study channel covariance matrices through Toeplitz theory, and characterize the spectral properties of the spectral density functions. The proposed MD-ACS is detailed in Section IV, including the review of the original ACS, the matrix-weight graph representation, and the NIP formulation with a greedy algorithm. The numerical results can be found in Section V, followed by the Conclusion in Section VI.

**Notation:** We use \( x, \mathbf{x}, \) and \( \mathbf{X} \) to represent scalar, vector, and matrix, respectively. For any scalar \( x \), we denote \( \{ x_n \}_{n=1}^{N} \triangleq \{ x_1, x_2, \ldots, x_N \} \). For the integer \( N \), we denote \( [N] \triangleq \{ 1, 2, \ldots, N \} \). A matrix \( \mathbf{X} \) is Hermitian if and only if \( \mathbf{X} = \mathbf{X}^H \), where \( \mathbf{X}^H \) is the conjugate transpose of \( \mathbf{X} \). \( \text{tr}(\mathbf{X}) \) denotes the trace of a matrix \( \mathbf{X} \). \( \mathbb{E}\{ \cdot \} \) denotes the expectation. The Kronecker and Hadamard products of two matrices \( \mathbf{X} \) and \( \mathbf{Y} \) are denoted by \( \mathbf{X} \otimes \mathbf{Y} \) and \( \mathbf{X} \odot \mathbf{Y} \), respectively. \( \mathcal{N}_C(\alpha, \beta) \) denotes the complex normal distribution, where \( \alpha \) and \( \beta \) are mean (vector) and variance (matrix), respectively. \( \mathbf{I}_M \) is the \( M \times M \) identity matrix, and \( \mathbf{F}_M \) is the discrete Fourier transform (DFT) matrices with \( [\mathbf{F}_M]_{p,q} = \frac{1}{\sqrt{M}} e^{-j 2\pi (p-1)(q-1)/M} \) for all \( p \in [M], q \in [M] \).

**II. CHANNEL AND SIGNAL MODEL**

**A. DP-UPA Channel Model**

We consider a single-cell massive MIMO system where the base station is equipped with an \( M_x \times M_y \times 2 \) dual-polarized uniform planar array (DP-UPA) serving \( N_U \) single-polarized single-antenna users. The DP-UPA consists of in total \( M = 2 M_x M_y \) antenna elements with \( M_x \) ports in each column and \( M_y \) ports in each row, and for each port there are two polarized antenna elements. According to 3GPP TR-36.873 [20], which is also referred by e.g., [21] and [22], the channel vector \( \mathbf{h} \) of DP-UPA can be represented as

\[
\mathbf{h} = \begin{bmatrix} \mathbf{h}_V \\ \mathbf{h}_H \end{bmatrix} \in \mathbb{C}^{M \times 1}
\]

where \( \mathbf{h}_V \in \mathbb{C}^{M_x \times 1} \) and \( \mathbf{h}_H \in \mathbb{C}^{M_y \times 1} \) correspond to the channel between the vertical (V)/horizontal (H) antenna and the user, respectively. For notational simplicity, let \( q \in \{ V, H \} \). Given the angle intervals of azimuth \( \mathcal{A} \) and elevation \( \mathcal{B} \), according to the channel model of 3GPP [20], the \( q \)-th sub-channel vector can be written as

\[
\mathbf{h}_q = \int_{\mathcal{B}} \int_{\mathcal{A}} \beta_q(\theta, \phi) \gamma_q a(\theta, \phi) d\theta d\phi
\]

where \( \mathcal{A} = [\theta_{\text{min}}, \theta_{\text{max}}] \), \( \mathcal{B} = [\phi_{\text{min}}, \phi_{\text{max}}] \) and \( |\mathcal{A}| = 2\delta_{\theta} \) and \( |\mathcal{B}| = 2\delta_{\phi} \), in which \( \delta_{\theta} \) and \( \delta_{\phi} \) are the angular spread (AS) of azimuth and elevation, respectively; \( \beta_q(\theta, \phi) \sim \mathcal{N}_C(0, \beta_q) \) denotes
the complex gain that is independent and identically distributed (i.i.d.) across paths; $\gamma_q$ is the polarization factor of the $q$-th sub-channel; and $\mathbf{a}(\theta, \phi)$ is the steering vector of DP-UPA antenna that possesses the same structure as that of UPA, and it can be written as [20] [21] [23]

$$\mathbf{a}(\theta, \phi) = \mathbf{a}_y(\theta, \phi) \otimes \mathbf{a}_x(\theta, \phi) = \begin{bmatrix}
1 \\
\exp\left(\frac{-2\pi d_y}{\lambda_w} \sin(\phi) \sin(\theta)\right) \\
\vdots \\
\exp\left(\frac{-2\pi d_y (M_y - 1)}{\lambda_w} \sin(\phi) \sin(\theta)\right)
\end{bmatrix} \otimes \begin{bmatrix}
1 \\
\exp\left(\frac{-2\pi d_x}{\lambda_w} \sin(\phi) \cos(\theta)\right) \\
\vdots \\
\exp\left(\frac{-2\pi d_x (M_x - 1)}{\lambda_w} \sin(\phi) \cos(\theta)\right)
\end{bmatrix} \tag{3}
$$

where $d_x$ and $d_y$ are antenna spacing of column and row array respectively, and $\lambda_w$ is the carrier wavelength.

\section*{B. Downlink Training and Precoding}

In this paper, we follow the comprehensive framework proposed in [16, Figure 4], which consists of (1) uplink pilot transmission, (2) uplink covariance estimation, (3) uplink-downlink covariance transformation, (4) downlink pilot transmission, (5) feeding back pilot measurements, (6) downlink channel estimation, and (7) downlink beamforming. As our focus in this paper is on the downlink precoding/beamforming, we assume the availability of downlink covariance matrix at the base station via the above steps (1)-(3). In what follows, we briefly reiterate the procedure of (4)-(7) to maintain certain level of self-containedness.

1) \textit{Downlink Pilot Transmission}: As in [16], the base station sends a space-time pilot matrix $\mathbf{S} \in \mathbb{C}^{T \times M'}$ to all users through a sparsifying precoder $\mathbf{V}_h \in \mathbb{C}^{M \times M'}$, where $T$ is the number of time slots used for pilot transmission, $M' \leq M$ is the dimension after the active sparsification, and the columns of $\mathbf{V}_h$ are chosen from an orthogonal matrix that will be specified later. As such, the received pilot signal $\mathbf{y}_i^p$ of $i$-th user can be written as

$$\mathbf{y}_i^p = \mathbf{SV}_h^\dagger \mathbf{h}_i + \mathbf{n}, \tag{4}$$

where $\mathbf{h}_i \in \mathbb{C}^{M \times 1}$ is the downlink channel vector of $i$-th user, and $\mathbf{n} \sim \mathcal{N}_\mathbb{C}(0, \sigma^2 \mathbf{I}_M)$ is the additive white Gaussian noise (AWGN). The pilot matrix $\mathbf{S}$ is up to design, subject to a total power constraint $\text{tr}(\mathbf{SV}_h^\dagger \mathbf{V}_h \mathbf{S}^\dagger) \leq \rho^p T$, where $\rho^p$ is the pilot signal power in each time slot.

2) \textit{Feeding Back Pilot Measurements}: For simplicity, we assume the users feed back their pilot signals $\mathbf{y}_i^p \in \mathbb{C}^{T \times 1}$ to the base station in an analog form. The digital feedback with quantization can be implemented according to well-developed techniques (see [24] and references therein). Due to possible user selection, only the selected users are required to send the pilot signals back
to the base station. In doing so, the base station could successfully acquires the perfect pilot signals \( \{ y_i^p \}_{i \in S} \) with \( S \) being the subset of selected users, which will be specified later.

3) **Downlink Channel Estimation:** Given the \( T \times 1 \) pilot signal \( y_i^p \), we aim to recover the \( M \times 1 \) channel vector \( h_i \) with \( M > T \), relying on the sparsity of \( h_i \) in the angular domain. Following the footstep in [16], we obtain the estimated channel vector via MMSE estimators as

\[
\hat{h}_i = R_{h,i} R_{y,i}^{-1} y_i^p,
\]

where \( R_{h,i} = \mathbb{E} \{ h_i (y_i^p)^H \} = R_i V_i S^H, \) \( R_{y,i} = \mathbb{E} \{ y_i^p (y_i^p)^H \} = S V_i^H R_i V_i S^H + \sigma^2 I \) with \( R_i \triangleq \mathbb{E} [ h_i h_i^H ] \) being the downlink channel covariance matrix of user-\( i \).

4) **Downlink Precoding:** With channel estimates, the base station transmit users’ data \( \{ d_i \}_{i \in S} \) through sparsifying precoders \( p_i \in \mathbb{C}^{M \times 1} \) for each selected user \( i \in S \). Thus, the received signal under FDD DP-UPA downlink data phase \( y_i^d \) of \( i \)-th user can be written as

\[
y_i^d = h_i^p p_i d_i + \sum_{j \in S \setminus i} h_i^p p_j d_j + n_i
\]

where \( n_i \sim \mathcal{N}_\mathbb{C}(0, \sigma_i^2) \) is the AWGN, and the sparsifying precoder \( p_i \) will be specified later. As the downlink covariance matrix estimation has been extensively investigated in the literature (e.g., [9], [16], [22]), we place our focus instead on designing the downlink precoder assuming that the downlink channel covariance matrix \( \{ R_i \}_{i=1}^{N_U} \) is perfectly known at the base station.

### III. Spectral Properties of Covariance Matrix

While some existing works have mentioned the Toeplitz structure of covariance matrices for ULA/UPA massive MIMO (e.g., [3], [25]), the extension to DP-UPA has not been fully understood. In what follows, we will inspect the structural properties of downlink channel covariance matrices \( \{ R_i \}_{i=1}^{N_U} \) for DP-UPA massive MIMO through the lens of Toeplitz matrix theory.

#### A. Toeplitz Matrix Theory

Before proceeding further, we first introduce the definitions related to Toeplitz matrix [26] and its extension to block Toeplitz matrix [19] and doubly Toeplitz matrix [27], [28].

Given a sequence of scalars \( \{ t_{-n+1}, \ldots, t_{-1}, t_0, t_1, \ldots, t_{n-1} \} \), an \( n \times n \) matrix \( T_n \) is a Toeplitz matrix if \( T_n(i,j) = t_{i-j} \) for all \( i,j \in [n] \). Similarly, given a sequence of \( M_1 \times M_2 \) matrices \( \{ T_{-n+1}, \ldots, T_{-1}, T_0, T_1, \ldots, T_{n-1} \} \), an \( n M_1 \times n M_2 \) matrix \( B_n \) is a block Toeplitz matrix if the \((i,j)\)-th \( M_1 \times M_2 \) submatrix \( [B_n](i,j) = T_{i-j} \) for all \( i,j \in [n] \). In particular, if \( T_m \) is an \( N \times N \)
Toeplitz matrix for all $-n+1 \leq m \leq n-1$, then $B_n$ is an $nN \times nN$ doubly Toeplitz matrix, also known as Toeplitz-block-Toeplitz (TBT) matrix (i.e., block Toeplitz matrix with Toeplitz blocks). Further, if $T_m$ is an $NM_1 \times NM_2$ block Toeplitz matrix for all $-n+1 \leq m \leq n-1$, then $B_n$ becomes an $nNM_1 \times nNM_2$ doubly block Toeplitz matrix. It can be viewed as a Toeplitz matrix with each element being block Toeplitz matrices, or a doubly Toeplitz matrix with each element being a general matrix. Throughout this paper, we consider Hermitian matrix, that is $t_{-i} = t_i^*$ for Toeplitz matrix and $T_{-i} = T_i^H$ for block Toeplitz matrix.

The circulant, block circulant, doubly circulant, doubly block circulant matrices can be similarly defined as their Toeplitz counterparts, where the only difference is the circular operation using the modulo operator $\text{mod}$, i.e., for the scalar sequence $[C_n]_{i,j} = c_{(i-j) \mod n}$ for all $i, j \in [n]$ and for the matrix sequence $[B_n]_{i,j} = C_{(i-j) \mod n}$ for all $i, j \in [n]$. Apparently, the (doubly block) circulant matrix is the special case of (doubly block) Toeplitz matrix. Given an $n \times n$ circulant matrix $C_n$, it can be diagonalized by DFT matrix, i.e., $C_n = F_n \Lambda F_n^H$ with $\Lambda$ being a diagonal matrix. For an $nN \times nN$ doubly circulant matrix, it can be diagonalized by 2D-DFT matrix $F_n \otimes F_N$. For an $NM_1 \times NM_2$ block circulant matrix $B_n$, it can be block-diagonalized by $B_n = (F_n \otimes I_{M_1}) \Sigma (F_n \otimes I_{M_2})^H$ where $\Sigma$ is an $NM_1 \times NM_2$ block diagonal matrix, with each block being an $M_1 \times M_2$ matrix.

When $n$ tends to infinity, each Toeplitz matrix can be associated with a generating function, which is a continuous and periodic function [19], [26]–[28]. For instance, the Hermitian Toeplitz matrix $T_n$ can be generated by a real function $F : [-1/2, 1/2] \mapsto \mathbb{R}$, i.e., $F(\omega) = \sum_{k=-\infty}^{\infty} t_k e^{j2\pi k\omega}$. Similarly, the Hermitian block Toeplitz matrix $B_n$ can be generated by a matrix-valued real function $F : [-1/2, 1/2] \mapsto \mathbb{R}^{M_1 \times M_2}$, i.e., $F(\omega) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} T_{k_1,k_2} e^{j2\pi (k_1\omega_1 + k_2\omega_2)}$. Further, for the Hermitian doubly block Toeplitz matrix, it can be generated by $F : [-1/2, 1/2]^2 \mapsto \mathbb{R}^{M_1 \times M_2}$, i.e.,

$$F(\omega_1, \omega_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} T_{k_1,k_2} e^{j2\pi (k_1\omega_1 + k_2\omega_2)}.$$  \hfill (7)

The circulant counterparts share the same generating functions.

B. Spectral Properties

By leveraging the Toeplitz matrix theory, we inspect the spectral properties of channel covariance matrix through a function analysis perspective. In particular, instead of looking into the channel covariance matrix, we investigate its spectral density in the angular domain. This is underpinned by the following lemma.
Lemma 1. The channel covariance matrix $R$ of DP-UPA massive MIMO can be represented, subject to row/column permutation, as a Hermitian doubly block Toeplitz matrix $\hat{R}$, which can be asymptotically block diagonalized by an orthogonal matrix $V = F_{M_x} \otimes F_{M_y} \otimes I_2$, (8)
as $M_x, M_y \to \infty$, where $F_n$ is an $n \times n$ DFT matrix, and the block-diagonal submatrices are uniformly sampled from the matrix-valued spectral density function, i.e.,

$$\Sigma(\omega_1, \omega_2) = \sum_{m_1 = -M_y+1}^{M_y-1} \sum_{m_2 = -M_x+1}^{M_x-1} [\hat{R}]_{m_1, m_2} e^{j2\pi(m_1\omega_1 + m_2\omega_2)}$$

with $[\hat{R}]_{m_1, m_2}$ being a $2 \times 2$ submatrix of $\hat{R}$.

Proof. See Appendix A. \qed

Remark 1. The $2 \times 2$ matrix-valued spectral density function $\Sigma(\omega_1, \omega_2)$ over $(\omega_1, \omega_2) \in [-1/2, 1/2]^2$ is the generating function of the doubly block Toeplitz matrix $\hat{R}$. As row/column permutation does not change spectral properties, $\Sigma(\omega_1, \omega_2)$ is the spectral density function of channel covariance matrix $R$ over the two-dimensional angular domain $[-1/2, 1/2]^2$. Similar to the ULA massive MIMO, we can also transform the signals from spatial to angular domain to exploit possible (block) sparsity of the spectral density. The columns of the DFT-type orthogonal matrices have been widely used as the common basis for Toeplitz, block Toeplitz and TBT matrix in massive MIMO such as precoder design for ULA [16], [29] and UPA [25] array, and pilot decontamination [30]–[32].

Equipped with Lemma 1, we are able to inspect the spectra of channel covariance matrix $R$ through its spectral density function $\Sigma(\omega_1, \omega_2)$. As such, we come up with the sparsity properties of DP-UPA antennas in the angular domain, as in Theorem 1.

Theorem 1. The spectral density function $\Sigma(\omega_1, \omega_2)$ has a compact support over the two-dimensional frequencies $(\omega_1, \omega_2) \in [-1/2, 1/2]^2$, i.e.,

$$\Sigma(\omega_1, \omega_2) = 0, \text{ if } (\omega_1, \omega_2) \notin \left(-\frac{d}{\lambda_w} z_{1_{\text{max}}}, \frac{d}{\lambda_w} z_{1_{\text{min}}}\right) \times \left(-\frac{d}{\lambda_w} z_{2_{\text{max}}}, \frac{d}{\lambda_w} z_{2_{\text{min}}}\right)$$

(10)

where $z_{i_{\text{min}}}$ and $z_{i_{\text{max}}}$ depend on a fixed AOA $\theta_c, \phi_c$ and AS $\Delta_1, \Delta_2$.

Proof. See Appendix B. \qed
Fig. 1: The normalized spectra of covariance matrices for 128 ULA, 16 × 8 UPA, and 8 × 8 × 2 DP-UPA antennas.

Remark 2. Theorem 1 is a generalization of the compact properties of spectral density function from ULA reported in [3] to DP-UPA antenna configurations. In contrast with the ULA, the spectral density function of DP-UPA is 2 × 2 matrix-valued because of the dual-polarization. In addition, for UPA and DP-UPA antennas, the compact supports of \( \Sigma(\omega_1, \omega_2) \) could be more dispersed, thanks to the two-dimensional array. This enables UPA-type antennas to server more users without causing severe pilot contamination or multiuser interference.

Thanks to the high resolution of large-scale antenna arrays, the azimuth and elevation AoAs are usually limited within a narrow range [33], so that \( z_i^{\max} \) and \( z_i^{\min} \) are confined within small intervals in \([-1, 1]\). As such, the compact support only covers a limited range of frequency range, and thus the spectral density exhibits sparsity properties in the angular domain. To illustrate the above points, we plot the spectral density of covariance matrices of ULA, UPA, and DP-UPA with the same number of antennas, using channels generated by QuaDRiGa [33] (See Section V for the configurations). In particular, Figure 1 shows the normalized diagonal elements of DFT-diagonalized covariance matrices for \( 128 \times 1 \) ULA, \( 16 \times 8 \) UPA, and \( 8 \times 8 \times 2 \) DP-UPA, respectively. It can be observed that ULA has one single yet wide support, and UPA and DP-UPA have multiple narrow supports. Additionally, for DP-UPA, it exhibits the block support where the supports appear in pair, which agrees with the 2 × 2 matrix-value spectral density function.

IV. Multi-Dimensional Active Channel Sparsification

Inspired by the matrix-valued spectral density function, we extend the active channel sparsification to multi-dimensional scenarios, with a generalized optimization problem formulation.
In what follow, we first summarize the merit of active channel sparsification proposed in [15], followed by a matrix-weight graph representation, and finally we bridge the general optimization problem formulation to an existing problem in combinatorial optimization.

A. Preliminary: Active Channel Sparsification

For the sake of self-containedness, we briefly introduce the main idea of active channel sparsification proposed in [15], with the focus on ULA antenna geometry.

1) Channel Representation: Each user’s channel is represented as a weighted sum of a set of vectors chosen from common bases. These common basis vectors are also referred to as virtual beams in the angular/beam domain. Usually, for ULA antenna setting, the columns of discrete Fourier transform (DFT) matrix are adopted as common basis vectors. This is underpinned by the fact that the channel covariance matrix of ULA antenna is a Toeplitz matrix, which asymptotically approximates the circulant matrix that can be diagonalized by DFT matrix.

Such a representation ensures that all users are represented in the same vector space, such that beam selection can be alternatively done by switching on/off the basis vectors. If we use $x_m$ to denote the status of the beam-$m$, we have

$$x_m = \begin{cases} 
1, & \text{if beam-}m \text{ is selected}, \\
0, & \text{otherwise}. 
\end{cases}$$

In a similar way, we can also impose a binary variable $y_i$ to denote user selection, i.e.,

$$y_i = \begin{cases} 
1, & \text{if user-} i \text{ is selected}, \\
0, & \text{otherwise}. 
\end{cases}$$

Hence, after adopting the user and beam selection strategy, the estimate of $i$-th user’s channel $h_i$, as in (5) in Section II-B, can be asymptotically written as

$$\hat{h}_i \approx \sum_{m=1}^{M} x_m \iota_{i,m} \sqrt{w_{i,m}} f_m$$

if user-$i$ is selected, i.e., $y_i = 1$, where $f_m$ is the $m$-th common basis vectors used for channel representation, $w_{i,m}$ is the corresponding coefficient that can be estimated by downlink training, and $\iota_{i,m}$ is a random variable. Usually, in the ULA setting, $f_m$ comes from the columns of the DFT matrix $F_M$ (e.g., [15], [34]), and in the $M_x \times M_y \times 2$ DP-UPA setting, $f_m$ is usually from the columns of the common basis $I_2 \otimes F_{M_x} \otimes F_{M_y}$ (e.g., [16]). Accordingly, the sparsifying precoder $V_h$ in (4) can be specified as the collection of basis vectors $\{f_m : x_m = 1\}$. 
2) Graph Representation: To describe the interaction between beam and user selection, we can construct a weighted bipartite graph, where beams are on one side and users are on the other side, and a beam and a user is connected if the beam contributes to channel representation of such user. By such bipartite graph representation, we establish the user-beam association with respect to the weighted combinations of channel representation.

For the readers’ reference, we introduce some graph definitions. Consider an undirected bipartite graph $G = (U, V, E)$ with two vertex sets $U$ and $V$, and an edge set $E$. For any $u \in U$ and $v \in V$, $e = (u, v) \in E$ if and only if $u$ and $v$ are connected with an edge $e$. The neighborhood of a vertex $v$ is the set of nodes $u \in U$ such that $(u, v) \in E$, i.e., $N_G(v) \triangleq \{ u \in U : (u, v) \in E \}$. The degree of a vertex $v$ is the number of nodes in the neighborhood of $v$, i.e., $\deg_G(v) \triangleq |N_G(v)|$ where $|N|$ is the cardinality of the set $N$. The adjacency matrix $A$ of the bipartite graph $G = (U, V, E)$ is a binary matrix, where $A_{i,j} = 1$ if $(i, j) \in E$ and 0 otherwise. The Bipartite matching of the bipartite graph $G = (U, V, E)$ is a subset of edges $M_G \subset E$ such that there are not edges in $M_G$ sharing the same vertex.

3) Beam/user Selection: The aim of beam/user selection is to switch on/off beams and users to avoid beam overlapping among selected users, in order to achieve the maximum multiplexing gain (i.e., prelog of the sum rate expression). The optimization problem was given in [15] as

\[
(P_1) : \max |M_{G'}| \tag{14a}
\]

s.t. \[ \deg_{G'}(u_i) \leq T, \quad \forall u_i \in U' \tag{14b} \]

\[ \sum_{b_m \in N_{G'}(u_i)} w_{i,m} \geq P, \quad \forall u_i \in U' \tag{14c} \]

where $|M_G|$ is the maximum cardinality bipartite matching number of the selected subgraph $G' = (B', U', E')$, and the degree constraint (14b) guarantees that for each the selected user $u_i \in U'$ the number of beams to represent this user’s channel vector is no more than $T$, and the power constraint (14c) is to ensure that for each selected user $u_i \in U'$ the sum power of representing beams is no less than $P$. The degree and power constraints ensure that each selected user should have a sufficient number of (but not too many) representing beams selected, so that those beams with little contribution to a user’s channel representation can be switched off.

As usually there are much more beams than users, by intuition, the maximum cardinality bipartite matching tends to select all users and only the users have severe conflicting representing beams will be unselected. As such, there is only implicit user selection through beam selection.
4) Casting as an MILP: By establishing the equivalence between the multiplexing gain of the users’ effective channel and the maximum cardinality bipartite matching of the graph representation, the objective of ACS can be solved by finding the solutions to an MILP [15] involving two sets of binary variables \( \{x_m\}_{m=1}^M \) and \( \{y_i\}_{i=1}^N \), and a set of continuous ones, \( \{z_{i,m}\}_{i=1,m=1}^{NU,M} \) i.e.,

\[
\begin{align*}
\mathcal{P}_1^' : \quad & \max_{x_m, y_i, z_{i,m}} \sum_{b_m \in B} \sum_{u_i \in U} z_{i,m} \\
\text{s.t.} \quad & z_{i,m} \leq [A]_{i,m}, \quad \forall b_m \in B, \; u_i \in U \\
& \sum_{u_i \in U} z_{i,m} \leq x_m, \quad \forall b_m \in B \\
& \sum_{b_m \in B} z_{i,m} \leq y_i, \quad \forall u_i \in U \\
& \sum_{b_m \in B} [A]_{i,m} z_{i,m} \leq Ty_i + M(1 - y_i), \quad \forall u_i \in U \\
& Py_i \leq \sum_{b_m \in B} [W]_{i,m} x_m, \quad \forall u_i \in U \\
& x_m \leq \sum_{u_i \in U} [A]_{i,m} y_i, \quad \forall b_m \in B \\
& x_m, y_i \in \{0, 1\} \quad \forall u_i \in U, \; b_m \in B \\
& z_{i,m} \in [0, 1] \quad \forall u_i \in U, \; b_m \in B
\end{align*}
\]

where binary matrix \( A \) is the adjacency matrix of graph \( G \), and \([W]_{i,m} = w_{i,m}\) indicates the contribution of the block beam \( m \) to the \( i \)-th user’s channel representation. By such an MILP formulation, we can adopt off-the-shelf solvers to find a feasible solution \( \{x^*_m\}_{m=1}^M \), \( \{y^*_i\}_{i=1}^N \) and \( \{z^*_{i,m}\}_{i=1,m=1}^{NU,M} \) efficiently, where the selected beams and users are indicated by \( \{m : x^*_m = 1\} \) and \( \{i : y^*_i = 1\} \) respectively in the optimal solution yielded by the MILP.

B. Matrix-weight Bipartite Graph Representation

From Section III, the covariance matrix \( \hat{R}_i \) can be asymptotically block-diagonalized by

\[
\lim_{M_y, M_x \to \infty} \hat{R}_i = (F_{M_y} \otimes F_{M_x} \otimes I_2) \Sigma_i (F_{M_y} \otimes F_{M_x} \otimes I_2)^H
\]

\[
= \sum_{m_1=1}^{M_y} \sum_{m_2=1}^{M_x} (f_{v,m_1} \otimes f_{h,m_2} \otimes I_2) \Sigma_i (m_1, m_2) (f_{v,m_1} \otimes f_{h,m_2} \otimes I_2)^H
\]
Fig. 2: Matrix-weight bipartite graph for channel representations, where the virtual block beams are denoted by a square with crossed lines (cf. cross-polarized antenna elements), the users are denoted by triangles, and the weights between beams and users \( E_{i,m} \) are \( 2 \times 2 \) matrices. (a) Channel representations from different users are overlapping in the sense that they share some common block beams (indicated by red edges) to represent their respective channels. (b) After active channel sparsification applied, some block beams (marked in gray) and users (marked in black) are switched off to avoid channel overlapping, so that the remaining users are not overlapping on active block beams.

where \( f_{v,m} \) and \( f_{h,m} \) are the \( m \)-th column of DFT matrices \( F_{My} \) and \( F_{Mx} \), respectively, and \( \Sigma_i(m_1, m_2) \) is the \( (M_y(m_1 - 1) + m_2) \)-th diagonal block matrix of \( \Sigma_i \).

Instead of using a vector to represent a virtual beam in the ULA and UPA settings, here we use a \( M \times 2 \) submatrix \( V_{m_1, m_2} \equiv f_{v,m_1} \otimes f_{h,m_2} \otimes I_2 \) to represent a virtual cross-polarized block beam. Similarly, we can represent all users’ channels by a bipartite graph with matrix-valued weights, where the cross-polarized block beams \( \{V_{m_1, m_2}, m_1 \in [M_y], m_2 \in [M_x]\} \) on one side and the users on the other side, and the beams and users are connected with edges of matrix-valued weights \( \{\Sigma_i(m_1, m_2), m_1 \in [M_y], m_2 \in [M_x]\} \). For notational simplicity, we use \( [\Sigma_i]_m \) to denote the matrix-valued weight for \( m \in [M/2] \) corresponding to some \( (m_1, m_2) \).

We refer to the scalar-weight graph representation of previous ACS formulation as single-dimension, and the matrix-weight one as multi-dimension bipartite graph representation. In particular, we represent the users’ channel covariance matrices in respect of the block beams in a matrix weighted bipartite graph \( G \) as in Fig. 2, where a block beam corresponds to a pair of cross-polarized antennas. For notational simplicity, we index the block beams as \( m \in [M/2] \). We define the matrix weighted bipartite graph \( G = (B, U, E) \), in which the block beams \( b \in B \) is on one side and users \( u \in U \) on the other side. Therefore, a beam \( b_m \) and a user \( u_i \) are connected with an edge \( (b_m, u_i) \in E \) if \( A_{i,m} = 1 \). It is worth noting that, the weight of edges \( (b_m, u_i) \in E \), i.e., \( E_{i,m} = [\Sigma_i]_m \) with \( m \) corresponding to some \( (m_1, m_2) \), is a \( 2 \times 2 \) matrix rather than a scalar.

With the block beam and user selection parameters \( x_m \) and \( y_i \), the estimated channel of \( i \)-th user,
as in (5) in Section II-B, can be approximately written as

\[ \hat{h}_i \approx \sum_{m=1}^{M/2} x_m V_m \left( \left[ \Sigma_i \right]_m \right)^{\frac{1}{2}} \iota_{i,m} \]  

(18)

where \( V_m \in \mathbb{C}^{M \times 2} \) corresponds to the block basis vectors \( V_{m_1, m_2} \) with \( m \) corresponding to some \((m_1, m_2)\), and \( \iota_{i,m} \in \mathbb{C}^{2 \times 1} \) is a random vector. Similarly, the sparsifying precoder \( V_h \) in (4) can be specified as the collection of \( \{V_m : x_m = 1\} \).

Let us explain the physical meaning of the matrix weighted bipartite graph. Each block beam is illustrated as a pair of crossed lines, in which the cross with red and black lines corresponds to the vertical and horizontal polarization antennas in the DP-UPA array, respectively. As a matter of fact, such a correspondence is resulted from the block-diagonalization of the channel covariance matrix, where it combines the cross-polarized antennas at the same position. The diagonal elements of \( E_{i,m} \in \mathbb{C}^{2 \times 2} \) represent the channel characteristics of the corresponding antenna, while the off-diagonal elements indicate the channel correlation between the vertical and horizontal antennas due to their cross-polarization. In Fig. 2(a), the edges in red indicate the inter-user spectral correlation between users’ channels in the angular domain, which results in potential inter-user interference for multi-user transmission. Fig. 2(b) presents a simple beam and user selection to reduce the possible beam overlapping for activated users. When actively switching off some block beams and users, the partial channel correlation of the remaining users is reduced, for which there is not any overlap on the activated beams anymore. Note here that the original channels covariance matrices are partially represented by the active block beams only.

While this may result in partial channel estimation and exploitation, it is expected not to degrade the overall multi-user performance as long as a proper beam and user selection strategy is designed. For the scalar case, it has been evidenced in [15] that the beam and user selection by ACS could improve overall performance over the ones without channel sparsification. In what follows, before proceeding with the matrix-weight bipartite graph representation, we take a step back to propose a more general formulation of ACS from the lens of combinatorial optimization.

C. Generalized Multi-dimensional Active Channel Sparsification (MD-ACS)

Given the above matrix-weight graph representation, we reformulate the original ACS [15] in a more general way. The generalization lies in two aspects: one is to extend one-to-one matching to many-to-many matching, the other one is to generalize scale-weight (i.e., single-dimensional)
to matrix-weight (i.e., multi-dimensional) matching with rate maximization and interference mitigation embedded instead of maximizing multiplexing gain.

1) From One-to-one to Many-to-Many Matching: In the original formulation in (14) of ACS, a subgraph $\mathcal{G}'$ is selected with active beams $\mathcal{B}'$ and users $\mathcal{U}'$, and the maximal bipartite matching is constructed in the induced subgraph $\mathcal{G}' = (\mathcal{B}', \mathcal{U}', \mathcal{E}')$. It has been shown in [15] that the cardinality of the maximal bipartite matching is equal to the multiplexing gain of multi-user transmission. If we take a step back, instead of working on the maximal (one-to-one) bipartite matching in the selected graph, we could consider the many-to-many matching on the original bipartite graph $\mathcal{G}$, where a number of beams can be associated to one user, and likewise each beam can serve multiple users as long as inter-user interference is properly controlled. As such, a more general formulation can be given by

$$\begin{align*}
\mathcal{P}_2 : \quad & \max \ w(M^*_G) \\
\text{s.t.} \quad & \deg_G(u_i) \leq \kappa_{b,i}, \quad \forall u_i \in \mathcal{U} \\
& \deg_G(b_m) \leq \kappa_{u,m}, \quad \forall b_m \in \mathcal{B}
\end{align*}$$

(19a)
(19b)
(19c)

where $M^*_G$ is the set of many-to-many matching, which is a generalization of one-to-one matching, $\kappa_{b,i} \leq T/2$ is the maximum beams can be assigned to user $i$ to guarantee that channel estimation is feasible [15], and $\kappa_{u,m}$ the maximum users that can reuse the same beam $m$ so that not much interference is caused one another. In contrast to the one-to-one matching, many-to-many matching allows each vertex on one side to be matched with multiple vertices on the other side.

The above many-to-many weighted matching is equivalent to the generalized multi-assignment problem (GMAP) [35], which is a generalized version of the assignment problem corresponding to one-to-one matching. The GMAP considers to assign a set of tasks to a set of agents. When a task is assigned to an agent, it produces profit and incurs cost. The aim of GMAP is to assign each task to multiple agents, where one agent can conduct multiple tasks, so that the total cost of all tasks is minimized and/or the total profit is maximized. Under the context of the multiple beam-user assignment, the above generalized ACS formulation can be reformulated as a GMAP with an integer programming as follows

$$\begin{align*}
\mathcal{P}'_2 : \quad & \max \sum_{i=1}^{N_u} \sum_{m=1}^{M/2} w_{i,m} z_{i,m} \\
\text{s.t.} \quad & \sum_{i=1}^{N_u} z_{i,m} \leq \kappa_u, \quad \forall b_m \in \mathcal{B}
\end{align*}$$

(20a)
(20b)
\[ \sum_{m=1}^{M/2} z_{i,m} \leq \kappa_u, \quad \forall u_i \in U \]  
\[ z_{i,m} \in \{0, 1\}, \quad \forall b_m \in B, \forall u_i \in U \]  

where \( z_{i,m} \) is a binary decision variable such that \( z_{i,m} = 1 \) indicates the \( m \)-th block beam is assigned to \( i \)-th user, and 0 otherwise; \( w_{i,m} \) is the corresponding profit for such an assignment, and it is a function of the matrix-weight \( E_{i,m} \). \( \kappa_u \) and \( \kappa_b \) are the maximum number of users and block beams to match each beam and user, respectively. For simplicity, we assume each user (resp. beam) is associated to the same number of beams (resp. users).

**2) From Single-dimensional to Multi-dimensional Matching:** The generalized formulation in (20) reduces the size of the integer program compared to (14) to a great extent, thanks to the GMAP formulation and the matrix-weight bipartite representation of beam-user association. However, the merits in the original formulation, e.g., multiplexing gain maximization in (15a) and interference control in (15f), are totally lost.

To remedy the above reformulation, in what follows, we integrate the consideration of sum rate maximization into the objective function, especially into the parameters \( \{w_{i,m}\} \), and relegate the interference control to a constraint. Such a remedy results in a nonlinear formulation, which motivates us to propose a greedy algorithm to solve it in an efficient way.

**a) Embedding Sum Rate Maximization and Interference Control:** Given the subset of selected users \( S = \{i : y_i^* = 1\} \), the achievable rate of \( i \)-th user with downlink precoder \( p_i \) can be written by

\[ R_i = \log \left( 1 + \frac{|h_i^H p_i|^2}{\sigma^2 + \sum_{j \in S \setminus i} |h_j^H p_j|^2} \right). \]

For the sake of tractability of optimization, we consider an asymptotic version of sum rate when \( M_x, M_y \to \infty \) so that the asymptotic zero-forcing precoder of \( i \)-th user can be simply written by the column vectors of common basis \( V \). In particular, we have

\[ p_i \in \mathcal{R}\{h_i\} \cap \mathcal{N}\{h_j, j \in S \setminus i\} \]

\[ = \{V_m : x_m \text{tr}([\Sigma_i]_m) \geq \delta, x_m y_j \text{tr}([\Sigma_j]_m) \leq \delta, \forall j \in [N_U] \setminus i, m \in [M/2]\} \]

where \( \mathcal{R}\{\cdot\} \) and \( \mathcal{N}\{\cdot\} \) are the range and null spaces of the subspace spanned by the vectors, and \( \delta \) is a threshold to determine if the block beam is strong enough to be considered.
Hence, with such asymptotic precoder, the asymptotic rate\(^1\) of \(i\)-th user can be written as

\[
R_i^\infty = \log \left( 1 + \frac{\text{tr} \left( \sum_{m=1}^{M/2} x_m \Sigma_i \Sigma_m^H \right)}{\sigma^2 + \text{tr} \left( \sum_{m=1}^{M/2} \sum_{j=1, j \neq i}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right)} \right) 
\]

\[
= \log \left( \sigma^2 + \text{tr} \left( \sum_{m=1}^{M/2} \sum_{j=1}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right) - \log \left( \sigma^2 + \text{tr} \left( \sum_{m=1}^{M/2} \sum_{j=1, j \neq i}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right) 
\]

\[
\geq \sum_{m=1}^{M/2} \log \left( \sigma^2 + \text{tr} \left( \sum_{j=1}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right) - \log \left( \sigma^2 + \text{tr} \left( \sum_{m=1}^{M/2} \sum_{j=1, j \neq i}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right)
\]  

(24)

where the first term is due to Jensen’s inequality with \(\log(\cdot)\) being a concave function, and the second term is due to an artificially introduced constraint

\[
\log \left( \frac{2\sigma^2}{M} + \text{tr} \left( \sum_{j=1, j \neq i}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right) \leq \sum_{m=1}^{M/2} \eta_{i,m}
\]

(25)

With Jensen’s inequality, the above constraint can be relaxed to

\[
\sum_{m=1}^{M/2} \log \left( \frac{2\sigma^2}{M} + \text{tr} \left( \sum_{j=1, j \neq i}^{N_U} y_j x_m \Sigma_i \Sigma_j^H \right) \right) \leq \sum_{m=1}^{M/2} \eta_{i,m}.
\]

(26)

Let us introduce two matrices \(P \in \mathbb{C}^{N_U \times \frac{M}{2}}\) and \(C \in \mathbb{C}^{N_U \times \frac{M}{2}}\) such that

\[
[P]_{i,m} = \log \text{tr} \left( \sum_{j=1}^{N_U} y_j \Sigma_i \Sigma_j^H \right),
\]

(28)

\[
[C]_{i,m} = \log \text{tr} \left( \sum_{j=1, j \neq i}^{N_U} y_j \Sigma_i \Sigma_j^H \right).
\]

(29)

The maximization of the asymptotic sum rate with joint user and beam selection can be approximately formulated in the following way

\[
(\mathcal{P}_3) : \max_{z_{i,m}} \sum_{m=1}^{M/2} \sum_{i=1}^{N_U} z_{i,m} [P]_{i,m}
\]

s.t. \((20b), (20c),\)

\[
[C]_{i,m} z_{i,m} \leq \eta_{i,m}, \forall b_m \in B, \forall u_i \in U
\]

(30c)

\[
z_{i,m} \in \{0, 1\}, \forall b_m \in B, \forall u_i \in U
\]

(30d)

\(^1\)We point out that the asymptotic rate here is with respect to the number of antennas, which is different from those at high SNR in the literature.
where the objective function (30a) comes from the lower bound of the asymptotic rate, with the constant parts dropped for simplicity, and the final constraint (30c) due to the constraint (27) to control interference, and \( z_{i,m} = x_m y_i \) is binary-valued.

The above optimization formulation can be recognized as a GMAP with an additional constraint (30c). The lower bound of the asymptotic rate can be regraded as the profits, and the constrained term in (27) can be treated as costs. As such, we refer to \( P \) and \( C \) as the profits and costs matrices, respectively. While the optimization problem (30) is linear for the parameters \( \{ z_{i,m} \} \), the profits and costs matrices \( P \) and \( C \) are dependent of user selection \( \{ y_j \} \), which is entangled with \( \{ z_{i,m} \} \) as \( z_{i,m} = x_m y_i \). This makes the problem a nonlinear integer program with respect to \( \{ x_m \} \) and \( \{ y_i \} \), which is challenging to solve. To overcome this, we propose a low-complexity greedy algorithm, avoiding overlaps between any two users in the matrix-weight bipartite graph.

b) Greedy Algorithm: As detailed earlier, given the channel covariance matrix \( \hat{R}_i \) with permuted rows and columns from the original one \( R_i \), we can construct a matrix-weight bipartite graph representation where the matrix weights \( [\Sigma_i]_m \) come from the block diagonalization of \( \hat{R}_i \). However, when it comes to the practical scenarios with a finite number of antennas, \( \hat{R}_i \) is not perfectly block-diagonalizable with the DFT matrix as in (16). To overcome this, a possible way is to approximate the matrix-weight \( [\hat{\Sigma}_i]_m \) by

\[
[\hat{\Sigma}_i]_m = [(F_{M_y} \otimes F_{M_x} \otimes I_2)^H \hat{R}_i (F_{M_y} \otimes F_{M_x} \otimes I_2)]_{m,m}
\] (31)

where \([ \cdot ]_{m,m}\) is the \( m \)-th \( 2 \times 2 \) block diagonal submatrix with \( m \in [M/2] \). It is readily verified that \( \lim_{M_x,M_y \to \infty} [\Sigma_i]_m = [\Sigma_i]_m \) for all \( i,m \). Thus, in what follows, we use \( [\hat{\Sigma}_i]_m \) instead of \( [\Sigma_i]_m \) for algorithm design in the practical scenarios.

For ease of presentation, we introduce a \( N_U \times M/2 \) matrix \( \Psi \) with \( \Psi_{i,m} = \text{tr}( [\hat{\Sigma}_i]_m ) \) to indicate the contribution of the \( m \)-th block-beam to the \( i \)-th user. Let us define a binary matrix \( A' \) for the greedy algorithm with elements specified as

\[
[A']_{i,m} = \begin{cases} 
1, & \text{if } m \in \text{max}_{n_p} \{ \Psi_{i,m'} \}, \forall m' \in [M/2] \\
0, & \text{Otherwise}
\end{cases}
\] (32)

where \( n_p \in [M/2] \) is a tunable integer parameter, and \( \text{max}_{n_p} \{ A \} \) returns the indices of the largest \( n_p \) values in the set \( A \). Here \( A' \) serve as a mask to filter out the insignificant weight matrices \( \{ [\Sigma_i]_m, \forall m \} \) and only keep \( n_p \) largest ones. In particular, if we set \( n_p = \kappa_b \), then after masking with \( A' \), there are at most \( \kappa_b \) block beams left that are connected to each user, so that the constraint (20c) is automatically satisfied.
For the greedy algorithm, according to the asymptotic analysis of sum rate, we define a specific evaluation function as

\[
\Phi(P, C) = \sum_i \sum_m x_m y_i (P_{i,m} - C_{i,m}), \tag{33}
\]

where \( P \) and \( C \) are profit and cost matrix as shown in (28) and (29), for which \( \hat{\Sigma}_{i,m} \) is used.

Given the bipartite graph representation \( G = (B, U, E) \) and the matrix-weight \( \hat{\Sigma}_{i,m} \) on the edge \((b_m, u_i)\), we propose a greedy algorithm to solve the optimization problem \((P_3)\). The detailed procedure is outlined in Algorithm 1.

Let us explain the greedy algorithm in detail. Instead of maximizing the profit with the cost as the constraint in (30), we define a new profit function as in (33) which takes both original profits and costs into account. At the beginning, each user \( i \) selects \( \kappa_b \) block beams with the largest weights as specified by \( A' \) in (32). This is to make the constraint (20c) automatically satisfied. Then, each block beam \( m \) determines whether the number of served users exceeds its capability \( \kappa_u \) to satisfy the constraint (20b). For those beams with more than \( \kappa_u \) users served that violate the constraint (20b), we need to determine if it is better to switch off this beam \( m \), or some users so that the constraint (20b) is satisfied. To make such a decision, we compute and compare two quantities \( \Phi_b \) and \( \Phi_u \) when either option is applied with respect to the newly defined profit in (33). This operation repeats till the constraint (20b) is satisfied for all active block beams. After each iteration, the weight matrix \( \Phi \) and the binary matrix \( A' \) will be updated, so that the deactivated users or beams will not be considered in the future. As such, the greedy algorithm results in a feasible solution after at most \( M^2 \) updates.

To summarize, compared with the original single-dimension ACS formulation in [15], [16], our proposed MD-ACS with greedy algorithm has the following advantages.

- While the original ACS is dedicated to the maximization of multiplexing gain, our proposed MD-ACS takes both sum rate maximization and interference control into account, which leads to better performance at finite SNR, as will be shown in Section V.
- In the original ACS, the same threshold is applied for all users and beams to construct the bipartite graph representation, and the resulting graph is sensitive to such threshold. In addition, there are quite a few tunable parameters in (15), which are challenging to fine-tune to arrive at the sweet spot for the optimal solution, so that an improper choice probably results in severe performance degradation. For our proposed MD-ACS, the integer-valued
Algorithm 1 Greedy Algorithm for Generalized MD-ACS

1: **Input:** $\{ \hat{R}_i, \forall i \}$, $\kappa_u, \kappa_b$

2: **Initialization:** $x_m = y_i = 1$ for all $i \in [N_U], m \in [M/2]$

3: Produce $2 \times 2$ diagonal submatrices $\{ [\hat{\Sigma}_i]_m \}_{i=1}^{N_U} \times \frac{M}{2}$ from $\{ \hat{R}_i \}_{i=1}^{N_U}$ according to (31)

4: Construct the bipartite graph representation $G = (B, U, E)$ with matrix-weights $\{ [\hat{\Sigma}_i]_m \}$ for the edge $(b_m, u_i) \in E$, and construct the weight matrix $\Psi$ with $[\Psi]_{i,m} = \text{tr}( [\hat{\Sigma}_i]_m )$

5: Compute profit and cost matrix $P, C$ as (28)-(29) with $[\hat{\Sigma}_i]_m$

6: Construct a binary matrix $A'$ as in (32) with $n_p = \kappa_b$, such that (20c) is satisfied

7: Update $\Psi$ as $\Psi \leftarrow A' \odot \Psi$, and set $M = \{ m : \sum_{i=1}^{N_U} x_m y_i [A']_{i,m} > \kappa_u, \forall m \in [M/2] \}$

8: while $M \neq \emptyset$ do

9: Select the beam $m \in M$

10: Compute (33) as $\Phi_b$ if the beam is switched off, i.e., $x_m = 0$

11: Compute (33) as $\Phi_u$ if only $\kappa_u$ users with the largest $[\Psi]_{i,m}$ are selected, i.e., $y_i = 0$ for all $i \notin \max^{\kappa_u} \{ [\Psi]_{j',m}, \forall i' \}$

12: if $\Phi_b > \Phi_u$ then

13: $x_m = 0$, and $[A']_{i,m} = 0, \forall i \in [N_U]$

14: else

15: $y_i = 0$, and $[A']_{i,m} = 0, \forall m \in [M/2], i \notin \max^{\kappa_u} \{ [\Psi]_{j',m}, \forall i' \}$

16: end if

17: Update $\Psi$ as $\Psi \leftarrow [A'] \odot \Psi$

18: Update $M \leftarrow \{ m : \sum_{i=1}^{N_U} x_m y_i [A']_{i,m} > \kappa_u, \forall m \in [M/2] \}$

19: end while

20: **Output:** $\{ x_m \}_{m=1}^{M/2}, \{ y_i \}_{i=1}^{N_U}$

parameters $\kappa_u$ and $\kappa_b$ are used to construct the bipartite graph, and the resulting graph is more flexible and suitable for greedy search.

- Due to the pre-determined bipartite graph representation and the implicit user selection, the original ACS is suitable to the homogeneous scenarios, whereas our proposed greedy algorithm is suitable for both homogeneous and heterogeneous scenarios (e.g., with both indoor and outdoor users), thanks to the adaptive bipartite graph construction and the explicit user selection, as will be demonstrated in Section V.
V. Numerical Results

In this section, we provide the numerical results of our proposed method — generalized multi-dimensional active channel sparsification (MD-ACS) — compared with the state-of-the-art ones in the practical DP-UPA FDD massive MIMO scenarios. The following baseline methods are considered for comparison.

- **No Selection**: All users and beams are activated.
- **JSDM**: A clustering algorithm that divides users into groups according to the similarity of their channel covariance matrices [3]. In each group, a user is randomly selected on behalf of the corresponding cluster. In JSDM, the number of clusters $K$ is essential, and is set to $K = \sum y^*_i$, where $y^*$ is the user selection vector obtained by our greedy algorithm.
- **ACS**: The original ACS on scale-weight bipartite graph representation, which was first proposed in [15] for ULA, and later on extended to DP-ULA in [16];
- **ACS-Matrix**: The conventional ACS with the MILP formulation on the matrix-weight bipartite graph representation, where the constraint (15f) is replaced by

$$Py_i \leq \sum_{b_m \in B} \text{tr} ([W]_{i,m}) x_m, \quad \forall u_i \in U. \quad (34)$$

- **Greedy Algorithm**: The proposed MD-ACS with greedy algorithm implementation as specified in Algorithm 1.

The downlink channel training and precoding follow the procedure in Section II-B, where the pilot matrix $S$ is a $T \times M'$ orthogonal matrix, with $M' = 2 \sum_{m=1}^{M/2} x_m \leq M$ being the number of activated virtual beams, and the average pilot signal power is set to $\rho^p = 1$.

For the simulation scenarios, we consider FDD downlink transmission in a single-cell massive MIMO network, where the base station is equipped with $M = M_x \times M_y \times 2$ DP-UPA antenna and serves $N_U$ single-antenna users. In order to evaluate the algorithms comprehensively and fairly, we adopt the QuaDRiGa channel model [33] to generate downlink channel vectors $h$. According to the 3GPP and the QuaDRiGa specifications [36], the Inter-Site Distance is set to be 500m and the ‘3GPP-3D-UMA’ scenario is considered. For the users’ located rules, the minimum distance from users to the base station is 10m. We choose 50% indoor and 50% outdoor users for downlink transmission, where the height of all the outdoor users is set to 1.5m. In all simulation scenarios, we assume the downlink channel covariance matrix is somehow available, which can be simply obtained by $R = \frac{1}{N} \sum_{t=1}^{N} h_t h_t^H$ using $N = 1000$ downlink channel vector realizations $h_t$ generated from QuaDRiGa, or obtained from uplink channel covariance matrix by
Fig. 3: Sum rate versus SNR with $4 \times 4 \times 2$ DP-UPA, $N_U = 15$ users and $T = 16$ timeslots.

Fig. 4: Sum rate versus pilot dimension with $4 \times 4 \times 2$ DP-UPA, $N_U = 15$ users and SNR = 20 dB.

Fig. 5: Sum rate versus SNR with $4 \times 4 \times 2$ DP-UPA, $N_U = 30$ users and $T = 16$ timeslots.

Fig. 6: Sum rate versus pilot dimension with $4 \times 4 \times 2$ DP-UPA, $N_U = 30$ users and SNR = 20 dB.

leveraging uplink-downlink reciprocity (e.g., [15]). Unless otherwise explicitly specified, for all the simulation scenarios, we choose the following parameters: $\kappa_b = 3, \kappa_u = 20$ for 64 antenna configuration and $\kappa_u = 12$ for 32 antenna configuration; pilot dimensions of training phase $T = 16$; for 32 antennas (Fig. 3-6), the antenna array is $M_x = M_y = 4$ and the FDD frame length is $T_c = 64$, and for 64 antennas (Fig. 7-11), $M_x = 4, M_y = 8$ and $T_c = 72$.

Figures 3 and 4 illustrate the sum rate of downlink transmission with $N_U = 15$ users in total versus SNR and the pilot dimension of the training phase, respectively. We can observe in Fig. 3 that, the proposed MD-ACS with greedy algorithm outperforms all other methods, and the gap is increasing as SNR goes. The ACS-like methods (i.e., ACS and ACS-Matrix) perform poorly at high SNR compared with No Selection, which is probably due to the fact
that the number of users for selection is quite limited so that activating all users may not be a bad idea. In Fig. 4, it appears the sum rate first increases as the pilot dimension does, because higher pilot dimension yields higher estimation accuracy of downlink channel, and therefore more accuracy downlink precoding. The sum rate is decreasing when pilot dimension increases further, because the more resource the training phase occupies, the less the transmission phase could use. For the ACS-like methods (i.e., ACS and ACS-Matrix), it looks too many users have been switched off, which results in severe performance degradation when $T$ is large. In Figures 5 and 6, $N_U = 30$ users are considered. It is observed that our proposed MD-ACS with greedy algorithm consistently outperform other methods. In this scenario, with a sufficiently large number of users, both ACS-Matrix and ACS perform better than JSDM and No Selection. The No Selection method confronts severe performance degradation – it is because there are too many users in the network, and user selection is crucial. In these simulations, ACS-Matrix outperforms the conventional ACS approach, which demonstrates the effectiveness of using matrix-weight bipartite graph representation. Notably, from Fig. 4 and Fig. 6, the optimal pilot dimensions that maximize the sum rate are different across algorithms. The optimal pilot dimension of ACS is around $T = 12$ while others are around $T = 16$. This suggests that ACS seems more dedicated to beam selection, while others (including the greedy algorithm) prefer user selection.

Further, in Fig. 7-10, we increase the number of antennas from 32 to 64, and consider two scenarios with $N_U = 30$ and $N_U = 60$ users. For the 64 DP-UPA antenna scenario with 30 users, compared with the 32-antenna case, the sum rate improvement of our greedy algorithm over other methods is diminishing, because the capability of serving more users is enhanced with more antennas, and thus user selection is not crucial. When the number of users increases from $N_U = 30$ to $N_U = 60$, we observe the same phenomenon as that in Figures 5 and 6. Interestingly, from Figures 8 and 10, we find that even if the number of antennas is increased, the optimal pilot dimension is still around $T = 16$ timeslots. It is worth noting that, ACS-Matrix with matrix-weight graph representation outperforms the conventional scalar-weight ACS method. It suggests that the matrix-weight formulation is more suitable than scalar-weight ACS for the DP-UPA scenario. As such, the improvement of our proposed greedy algorithm comes from two aspects: the matrix-weight MILP formulation and the search-based user/beam selection strategy.

Fig. 11 shows the sum rate versus the number of users in the cell when the pilot dimension is set to $T = 16$. We can observe that: (1) When the number of users is small, e.g., $N_U \leq 30$, user selection is unnecessary, because the sum rates are nearly the same for the proposed greedy
Fig. 7: Sum Rate versus SNR with $4 \times 8 \times 2$ DP-UPA, $N_U = 30$ users and $T = 16$ timeslots.

Fig. 8: Sum Rate versus Timeslots with $4 \times 8 \times 2$ DP-UPA, $N_U = 30$ users and SNR = 20 dB.

Fig. 9: Sum Rate versus SNR with $4 \times 8 \times 2$ DP-UPA, $N_U = 60$ users and $T = 16$ timeslots.

Fig. 10: Sum Rate versus Timeslots with $4 \times 8 \times 2$ DP-UPA, $N_U = 60$ users and SNR = 20 dB.

Fig. 11: Sum Rate versus the number of users with $4 \times 8 \times 2$ DP-UPA, $T = 16$ timeslots and SNR = 20 dB.
algorithm, compared with No Selection. (2) As the number of users increases, the benefit of user selection emerges, and it becomes crucial when the number of users is large, e.g., \( N_U \geq 45 \). (3) Our proposed MD-ACS with greedy algorithm always outperforms the conventional ACS thanks to the matrix-weight graph representation and the search-based greedy user/beam selection.

VI. CONCLUSION

In this work, we have investigated downlink sparsifying precoder design and user selection in DP-UPA FDD massive MIMO systems using active channel sparsification (ACS). By extending the original scalar-weight bipartite graph representation of user-beam association to a matrix-weight bipartite graph, we proposed a generalized multi-dimensional ACS (MD-ACS) for DP-UPA antenna configurations with a nonlinear integer program formulation. Inspired by the generalized multi-assignment problem, we proposed an efficient greedy algorithm to solve the nonlinear integer problem, and observed its superiority in extensive simulation results using QuaDriGa channel models. We believe such an improvement of the ACS methodology could pave the way for the potential deployment of ACS to the practical FDD massive MIMO systems.

APPENDIX

A. Proof of Lemma 1

Given the channel vector in (1), the covariance matrix \( R = \mathbb{E}\{hh^H\} \) can be written as

\[
R = \begin{bmatrix}
\mathbb{E}\{h_V h_V^H\} & \mathbb{E}\{h_V h_H^H\} \\
\mathbb{E}\{h_H h_V^H\} & \mathbb{E}\{h_H h_H^H\}
\end{bmatrix} \triangleq \begin{bmatrix}
R_1 & R_2 \\
R_2^H & R_3
\end{bmatrix}.
\]  

For \( k = 1, 2, 3 \) we have

\[
R_k = \int_{\Omega} p_k a(\theta, \phi) a^H(\theta, \phi) d\theta d\phi
\]

where \( \Omega = \{(\theta, \phi) : \theta \in \mathcal{A}, \phi \in \mathcal{B}\} \) and

\[
p_1 = \gamma_V \gamma_V^* \mathbb{E}\{\beta_V \beta_V^*\}, \quad p_2 = \gamma_V \gamma_H^* \mathbb{E}\{\beta_V \beta_H^*\}, \quad p_3 = \gamma_H \gamma_H^* \mathbb{E}\{\beta_H \beta_H^*\}
\]  

with \( \beta_V \) and \( \beta_H \) being vertical and horizontal polarization respectively. In fact, the submatrix \( R_k \) has the same structure as the covariance matrix of UPA, which is a doubly Toeplitz matrix. By letting \( P_\beta = \begin{pmatrix} p_1 & p_2^H \\
p_2 & p_3^H \end{pmatrix} \), the covariance matrix \( R \) can be alternatively written as

\[
R = P_\beta \otimes \int_{\Omega} a(\theta, \phi) a^H(\theta, \phi) d\theta d\phi = P_\beta \otimes B
\]
where
\[
B = \int_{\Omega} a_y(\theta, \phi) a^*_y(\theta, \phi) \otimes a_x(\theta, \phi) a^*_x(\theta, \phi) d\theta d\phi 
\]
\[
= \int_{\Omega} \begin{bmatrix}
B_{11} & \cdots & B_{1M_y} \\
\vdots & \ddots & \vdots \\
B_{M_y1} & \cdots & B_{M_yM_y}
\end{bmatrix}
\begin{bmatrix}
\end{bmatrix}
\]
\[
\text{(39)}
\]
For \(p, q \in [M_y]\), each block \(B_{pq}\) can be written as
\[
B_{pq} = [a_y(\theta, \phi) a^*_y(\theta, \phi)]_{pq} A(\theta, \phi) = e^{\frac{2\pi}{\lambda_w} d_y (p-q) \sin \phi \sin \theta} A(\theta, \phi)
\]
\[
\text{(41)}
\]
with
\[
[A(\theta, \phi)]_{ij} = [a_x(\theta, \phi)]_i [a^*_x(\theta, \phi)]_j = e^{\frac{2\pi}{\lambda_w} d_x (i-j) \sin \phi \cos \theta}.
\]
\[
\text{(42)}
\]
It appears that the elements in \(A(\theta, \phi)\) only depend on \((i-j)\) and the submatrices in \(B_{pq}\) only depend on \((p-q)\). Therefore, we conclude that \(B\) is a doubly Toeplitz matrix.

To facilitate the inspection from the perspective of generating function for Toeplitz matrices, we transform \(R\) into a doubly block Toeplitz matrix by row/column permutation. Following the footsteps in [37], we permutate \(R\) by a perfect shuffle matrix \(Q\) as
\[
\hat{R} = QRQ^T = QP_\beta \otimes BQ = B \otimes P_\beta
\]
\[
\text{(43)}
\]
with
\[
Q = \begin{bmatrix}
I_M \left( 1 : \frac{M}{2} : M, : \right) \\
I_M \left( 2 : \frac{M}{2} : M, : \right) \\
\vdots \\
I_M \left( \frac{M}{2} : \frac{M}{2} : M, : \right)
\end{bmatrix}
\]
\[
\text{(44)}
\]
By the permutation, \(\hat{R}\) is doubly Toeplitz matrix, that is, an \(M_x M_y \times M_x M_y\) doubly block Toeplitz matrix, with each element being a 2 \(\times\) 2 matrix. In particular, the \((m_1, m_2)\)-th submatrix \([\hat{R}]_{m_1, m_2}\) can be given by
\[
[\hat{R}]_{m_1, m_2} = \int_{\Omega} e^{\frac{2\pi}{\lambda_w} (d_y m_1 \sin \phi \sin \theta + d_x m_2 \sin \phi \cos \theta)} d\theta d\phi P_\beta.
\]
\[
\text{(45)}
\]
When \(M_x, M_y \to \infty\), it is known in [19] that the \((m_1, m_2)\)-th submatrix of \(\hat{R}\) can be given by
\[
[\hat{R}]_{m_1, m_2} = \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} \Sigma(\omega_1, \omega_2) e^{-j2\pi(m_1 \omega_1 + m_2 \omega_2)} d\omega_1 d\omega_2
\]
\[
\text{(46)}
\]
through its generating function

\[ \Sigma(\omega_1, \omega_2) = \sum_{m_1 = -\infty}^{\infty} \sum_{m_2 = -\infty}^{\infty} \hat{R}_{m_1, m_2} e^{2\pi i (m_1 \omega_1 + m_2 \omega_2)}. \] 

(47)

It is known that for any Toeplitz matrix \( T_n \), when \( n \to \infty \), there exists a circulant matrix \( C_n \) sharing the same generating function \([34]\). This applies to the extensions, e.g., doubly (block) Toeplitz and circulant matrices. It is known in \([19]\) that circulant matrix can be diagonalized by DFT matrix, and this can be extended to block and doubly block Toeplitz matrices. As such, for the doubly block Toeplitz matrix \( \hat{R} \), there exists a doubly block circulant matrix \( \hat{C} \) such that

\[ \hat{C} = (F_{M_x} \otimes F_{M_y} \otimes I_2) \Sigma(F_{M_x} \otimes F_{M_y} \otimes I_2)^H \]

(48)

where \( \Sigma \) is a block diagonal matrix with \( M_x M_y \) non-zero diagonal blocks of size \( 2 \times 2 \) each.

According to \([38, \text{Theorem 2}]\), the diagonal blocks of \( \Sigma \) is the uniform sampling of the generating function \( \Sigma(\omega_1, \omega_2) \) on the following grids

\[ (\omega_1, \omega_2) = \left( -\frac{1}{2} + \frac{m_1}{M_y}, -\frac{1}{2} + \frac{m_2}{M_x} \right), \quad \forall m_1 \in [M_y] - 1, \ m_2 \in [M_x] - 1. \]

(49)

B. Proof of Theorem 1

According to Lemma 1, by letting \( d_x = d_y = d \) and plugging (45) into the spectral density function \( \Sigma(\omega_1, \omega_2) \), we have

\[ \Sigma(\omega_1, \omega_2) = \sum_{m_1 = -\infty}^{\infty} \sum_{m_2 = -\infty}^{\infty} [\hat{R}]_{m_1, m_2} e^{2\pi i (m_1 \omega_1 + m_2 \omega_2)} \]

(50)

\[ = P_\beta \int_{\Omega} \sum_{m_1 = -\infty}^{\infty} \sum_{m_2 = -\infty}^{\infty} e^{2\pi i m_1 \left( \frac{d}{\lambda_w} \sin \phi \sin \theta + \omega_1 \right)} e^{2\pi i m_2 \left( \frac{d}{\lambda_w} \sin \phi \cos \theta + \omega_2 \right)} d\theta d\phi \]

(51)

\[ = P_\beta \int_{\Omega} \left( \sum_{m_1 = -\infty}^{\infty} e^{2\pi i m_1 \left( \frac{d}{\lambda_w} \sin \phi \sin \theta + \omega_1 \right)} \right) \left( \sum_{m_2 = -\infty}^{\infty} e^{2\pi i m_2 \left( \frac{d}{\lambda_w} \sin \phi \cos \theta + \omega_2 \right)} \right) d\theta d\phi \]

(52)

\[ = P_\beta \int_{\Omega} \left( \sum_{m_1 = -\infty}^{\infty} \delta \left( m_1 - \left( \frac{d}{\lambda_w} \sin \phi \sin \theta + \omega_1 \right) \right) \right) \]

\[ \cdot \left( \sum_{m_2 = -\infty}^{\infty} \delta \left( m_2 - \left( \frac{d}{\lambda_w} \sin \phi \cos \theta + \omega_2 \right) \right) \right) d\theta d\phi \]

(53)

where the last equation is due to Poisson Summation Formula \([39]\).

Further, let \( z_1 = \sin \phi \sin \theta \) and \( z_2 = \sin \phi \cos \theta \). Define \( z_i^{\text{max}} = \max_{\phi, \theta} \{ z_i \} \) and \( z_i^{\text{min}} = \min_{\phi, \theta} \{ z_i \} \). Due to the property of delta function, only if we have both \( \omega_1 = m_1 - \frac{d}{\lambda_w} z_1 \) and \( \omega_2 = m_2 - \frac{d}{\lambda_w} z_2 \), \( \Sigma(\omega_1, \omega_2) \) is a non-zero matrix. Given that \( m_1, m_2 \in \mathbb{Z}, -1 \leq z_i^{\text{min}} \leq z_i^{\text{max}} \leq 1 \),
and $\omega_1, \omega_2 \in \left(-\frac{1}{2}, \frac{1}{2}\right)$, the only possible integer of $m_1$ and $m_2$ is 0. Thus, the range of $\omega_i$ that yields non-zero $\Sigma(\omega_1, \omega_2)$ depends on that of $z_i^{\min}$ and $z_i^{\max}$, i.e. $\omega_i \in \left[-\frac{d}{\lambda_w} z_i^{\max}, \frac{d}{\lambda_w} z_i^{\min}\right], i = \{1, 2\}$. As such, given a set of AOA $\theta_c, \phi_c$ and AS $\Delta_1, \Delta_2$, we are able to obtain a compact support that is related to both elevation and azimuth AOAs. Even when the special points, $\omega_1, \omega_2 = \pm \frac{1}{2}$, are considered, such that $m_1, m_2 = \pm 1$ might exist, we only have the corresponding points under $z_i = \pm 1$ that does not alter the conclusion.

REFERENCES

[1] 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, 2014.

[2] E. Björnson, J. Hoydis, and L. Sanguinetti, “Massive MIMO networks: Spectral, energy, and hardware efficiency,” *Foundations and Trends in Signal Processing*, vol. 11, no. 3-4, pp. 154–655, 2017.

[3] A. Adhikary, J. Nam, J. Ahn, and G. Caire, “Joint spatial division and multiplexing—the large-scale array regime,” *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6441–6463, 2013.

[4] J. Nam, A. Adhikary, J. Ahn, and G. Caire, “Joint spatial division and multiplexing: Opportunistic beamforming, user grouping and simplified downlink scheduling,” *IEEE J. Sel. Topics in Signal Processing*, vol. 8, no. 5, pp. 876–890, 2014.

[5] X. Rao and V. K. Lau, “Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems,” *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3261–3271, 2014.

[6] Z. Gao, L. Dai, Z. Wang, and S. Chen, “Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO,” *IEEE Transactions on Signal Processing*, vol. 63, no. 23, pp. 6169–6183, 2015.

[7] Y. Ding and B. D. Rao, “Dictionary learning-based sparse channel representation and estimation for FDD massive MIMO systems,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 8, pp. 5437–5451, 2018.

[8] H. Xie, F. Gao, S. Jin, J. Fang, and Y. C. Liang, “Channel estimation for TDD/FDD massive MIMO systems with channel covariance computing,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4206–4218, 2018.

[9] L. Miretti, R. L. G. Cavalcante, and S. Stanczak, “FDD massive MIMO channel spatial covariance conversion using projection methods,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[10] S. Haghighatshoar, M. B. Khalilsarai, and G. Caire, “Multi-band covariance interpolation with applications in massive MIMO,” in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 386–390.

[11] C.-K. Wen, W.-T. Shih, and S. Jin, “Deep learning for massive MIMO CSI feedback,” *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.

[12] T. Wang, C.-K. Wen, S. Jin, and G. Y. Li, “Deep learning-based CSI feedback approach for time-varying massive MIMO channels,” *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 416–419, 2018.

[13] Y. Jang, G. Kong, M. Jung, S. Choi, and I. Kim, “Deep autoencoder based CSI feedback with feedback errors and feedback delay in FDD massive MIMO systems,” *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 833–836, 2019.

[14] M. Arnold, S. Dörner, C. Cammerer, S. Yan, J. Hoydis, and S. t. Brink, “Enabling FDD massive MIMO through deep learning-based channel prediction,” *arXiv preprint arXiv:1901.03664*, 2019.

[15] M. B. Khalilsarai, S. Haghighatshoar, X. Yi, and G. Caire, “FDD massive MIMO via UL/DL channel covariance extrapolation and active channel sparsification,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 121–135, 2018.

[16] M. B. Khalilsarai, T. Yang, S. Haghighatshoar, X. Yi, and G. Caire, “Dual-polarized FDD massive MIMO: A comprehensive framework,” *arXiv preprint arXiv:2008.11182*, 2020.
30

[17] M. B. Khalilsarai, T. Yang, S. Haghighatshoar, and G. Caire, “Structured channel covariance estimation from limited samples in massive MIMO,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.

[18] H. Liu, X. Yuan, and Y. J. Zhang, “Statistical beamforming for FDD downlink massive MIMO via spatial information extraction and beam selection,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 7, pp. 4617–4631, 2020.

[19] J. Gutiérrez-Gutiérrez, P. M. Crespo *et al.*, “Block Toeplitz matrices: Asymptotic results and applications,” *Foundations and Trends® in Communications and Information Theory*, vol. 8, no. 3, pp. 179–257, 2012.

[20] 3GPP, “Study on 3D channel model for LTE,” *Tech. Rep. 3GPP 36.873(V12.7.0)*, 2018.

[21] C. Qian, X. Fu, N. D. Sidiropoulos, and Y. Yang, “Tensor-based channel estimation for dual-polarized massive MIMO systems,” *IEEE Transactions on Signal Processing*, vol. 66, no. 24, pp. 6390–6403, 2018.

[22] L. Miretti, R. L. G. Cavalcante, and S. Stańczak, “Downlink channel spatial covariance estimation in realistic FDD massive MIMO systems,” in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2018, pp. 161–165.

[23] A. Lu, X. Gao, X. Meng, and X. Xia, “Omnidirectional precoding for 3D massive MIMO with uniform planar arrays,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2628–2642, 2020.

[24] D. J. Love and R. W. Heath, “Equal gain transmission in multiple-input multiple-output wireless systems,” *IEEE Transactions on Communications*, vol. 51, no. 7, pp. 1102–1110, 2003.

[25] H. Yu, L. You, W. Wang, and X. Yi, “Active channel sparsification for uplink massive MIMO with uniform planar array,” *to appear in IEEE Transactions on Wireless Communications*, 2021.

[26] R. M. Gray, *Toeplitz and circulant matrices: A review*., now publishers inc, 2006.

[27] P. A. Voois, “A theorem on the asymptotic eigenvalue distribution of Toeplitz-block-Toeplitz matrices,” *IEEE Transactions on Signal Processing*, vol. 44, no. 7, pp. 1837–1841, 1996.

[28] M. Oudin and J. P. Delmas, “Asymptotic generalized eigenvalue distribution of Toeplitz block Toeplitz matrices,” in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 3309–3312.

[29] X. Rao and V. K. N. Lau, “Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems,” *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3261–3271, 2014.

[30] H. Yin, D. Gesbert, M. Filippou, and Y. Liu, “A coordinated approach to channel estimation in large-scale multiple-antenna systems,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 264–273, 2013.

[31] L. You, X. Gao, X.-G. Xia, N. Ma, and Y. Peng, “Pilot reuse for massive MIMO transmission over spatially correlated Rayleigh fading channels,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 3352–3366, 2015.

[32] Z. Chen and C. Yang, “Pilot decontamination in wideband massive MIMO systems by exploiting channel sparsity,” *IEEE Transactions on Wireless Communications*, vol. 15, no. 7, pp. 5087–5100, 2016.

[33] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, “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, 2014.

[34] R. M. Gray *et al.*, “Toeplitz and circulant matrices: A review,” *Foundations and Trends® in Communications and Information Theory*, vol. 2, no. 3, pp. 155–239, 2006.

[35] J. S. Park, B. H. Lim, and Y. Lee, “A Lagrangian dual-based branch-and-bound algorithm for the generalized multi-assignment problem,” *Management Science*, vol. 44, no. 12-part-2, pp. S271–S282, 1998.

[36] “QuaDRiGa documentation.” [https://quadriga-channel-model.de](https://quadriga-channel-model.de).

[37] H. V. Henderson and S. R. Searle, “The vec-permutation matrix, the vec operator and Kronecker products: a review,” *Linear and Multilinear Algebra*, vol. 9, no. 4, pp. 271–288, 1981.

[38] X. Yi, “Asymptotic singular value distribution of linear convolutional layers,” *arXiv preprint arXiv:2006.07117*, 2020.

[39] A. Lapidoth, *A foundation in digital communication*. Cambridge University Press, 2017.