Joint Estimation of Azimuth and Elevation via Manifold Separation for Arbitrary Array Structures

Jie Zhuang, *Member, IEEE*, Chenghua Duan, Wei Wang, and Zhi Chen

***Abstract*—Direction-of-arrival estimation is a ubiquitous task in array processing. The conventional MUltiple SIgnal Classification (MUSIC) method is often computationally expensive, particularly in the application of joint azimuth and elevation estimation. In this paper, we propose an efficient way to compute the two-dimentional (2D) spatial spectrum. By using the manifold separation technique, we find that the 2-D null-spectrum has the form of discrete Fourier transform (DFT). Moreover, the coefficients of the DFT form can be truncated and computed in parallel. Then by exploiting the fact that the nonzero coefficients are concentrated only at the top-left corner, we present a partial 2-D DFT to compute the 2-D spatial spectrum, which can reduce the latency significantly. In order to improve the estimation performance, we formulate an iterative multidimensional-subspace-fitting estimator in which we apply the first-order Taylor expansion to expand the projection operator. At each iteration, the estimation errors can be computed by solving a set of linear equations. Numerical results demonstrate that the proposed estimator offers better estimation performance compared with the classical 2-D MUSIC.**

***Index Terms*—2D direction-of-arrival (DOA) estimation, arbitrary arrays, azimuth and elevation, partial 2D DFT, multidimensional subspace fitting, manifold separation technique (MST).**

I. Introduction

DIRECTION-OF-ARRIVAL (DOA) estimation is a ubiquitous task concerned in array processing in which a phased array is used to find the direction of a target by analyzing the phase difference among the array antennas without mechanical movement. This technique is attractive for the applications of automotive radars [1], vehicular position tracking and vehicle-to-vehicle radio channel characterization [2]. In [3], it is pointed out that the terahertz (THz) band may be envisioned as the next frontier for vehicular networks. Due to the THz propagation, a primarily line-of-sight (LOS) channel environment is highly likely and therefore the wireless channel can be modeled with the knowledge of DOA estimations. The DOA estimation problem that we will deal with in this paper can be briefly stated as follows: estimate the azimuth and elevation angles using arbitrary array configurations.

Let us consider an array of L antennas with arbitrary geometry operating in the presence of K narrowband signals. The signal vector received at the time instant t can be expressed as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{m}(t) + \mathbf{n}(t) \in \mathcal{C}^L \tag{1}$$

where $\mathbf{n}(t)$ represents the additive white Gaussian noise. The matrix $\mathbf{A}$ collects the steering vectors of the K signals, i.e.,

$$\mathbf{A} = [\mathbf{a}(\theta_1, \phi_1), \ldots, \mathbf{a}(\theta_K, \phi_K)] \in \mathcal{C}^{L\times K} \tag{2}$$

where $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi]$ denote the azimuth and elevation angles respectively. The vector $\mathbf{m}(t)$ has elements the K complex narrowband signal envelopes.

The theoretical covariance matrix of the received vector $\mathbf{x}(t)$ can be expressed as

$$\mathbf{R} = \mathbf{A}\mathbf{R}_s\mathbf{A}^H + \sigma_n^2\mathbf{I}_L \in \mathcal{C}^{L\times L} \tag{3}$$

where $\mathbf{R}_s = E\{\mathbf{m}(t)\mathbf{m}^H(t)\}$ is the signal covariance matrix, σ_n^2 is the noise power, and $\mathbf{I}_L$ denotes an L-by-L identity matrix. The notation $E\{\cdot\}$ represents the expectation. In practical applications, the covariance matrix can be estimated as follows

$$\widehat{\mathbf{R}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{x}(t)\mathbf{x}^H(t) \tag{4}$$

where $(\cdot)^H$ denotes the conjugate transpose operation and T is the snapshot number. Then we perform eigenvalue decomposition on $\widehat{\mathbf{R}}$ to produce

$$\widehat{\mathbf{R}} = \widehat{\mathbf{E}}_s\widehat{\mathbf{\Lambda}}_s\widehat{\mathbf{E}}_s^H + \widehat{\mathbf{E}}_n\widehat{\mathbf{\Lambda}}_n\widehat{\mathbf{E}}_n^H \tag{5}$$

where $\widehat{\mathbf{E}}_s \in \mathcal{C}^{L\times K}$ is the eigenvectors associated with the largest K eigenvalues and $\widehat{\mathbf{E}}_n \in \mathcal{C}^{L\times(L-K)}$ represents the eigenvectors corresponding to the remaining small eigenvalues. Commonly $\widehat{\mathbf{E}}_s$ and $\widehat{\mathbf{E}}_n$ are referred to as the signal-subspace eigenvectors and noise-subspace eigenvectors. The diagonal matrices $\widehat{\mathbf{\Lambda}}_s = \mathrm{diag}[\widehat{\lambda}_1, \ldots, \widehat{\lambda}_K]$ and $\widehat{\mathbf{\Lambda}}_n = \mathrm{diag}[\widehat{\lambda}_{K+1}, \ldots, \widehat{\lambda}_L]$ have diagonal elements associated with the signal and noise eigenvalues respectively, where the eigenvalues are arranged in decreasing order.

By exploiting the orthogonality between the signal and noise subspace, the well-known MUltiple SIgnal Classification (MUSIC) method [4] searches the continuous array manifold vector

Manuscript received August 5, 2017; revised November 21, 2017; accepted January 31, 2018. Date of publication February 5, 2018; date of current version July 16, 2018. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61571090, in part by the National Major Fund under Grant 6140518040101, and in part by the China Postdoctoral Science Foundation under Grant 2015M580785. The review of this paper was coordinated by Dr. C. Yuen. *(Corresponding author: Jie Zhuang.)*

J. Zhuang, C. Duan and W. Wang are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China (e-mail: jz@uestc.edu.cn; chduan_10@163.com; wangweiuestc@126.com).

Z. Chen is with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China (e-mail: chenzhi@uestc.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2018.2801785

over the area of θ and ϕ to find the K minima of the following null-spectrum cost function

$$(\widehat{\theta},\widehat{\phi}) = \arg\min_{\theta,\phi}\{\mathbf{a}(\theta,\phi)^H \widehat{\mathbf{E}}_n \widehat{\mathbf{E}}_n^H \mathbf{a}(\theta,\phi)\}. \tag{6}$$

However, such spectral search process may be unaffordable for some real-time implementations, which can be explained by the following fact. To obtain each search point, the Frobenius norm $\|\widehat{\mathbf{E}}_n^H \mathbf{a}(\theta,\phi)\|^2$ (or $\|\widehat{\mathbf{E}}_s^H \mathbf{a}(\theta,\phi)\|^2$) has to be computed. This drawback becomes particularly apparent in joint estimation of azimuth and elevation since we have to search over two dimensions.

In order to reduce the computational load of MUSIC methods, a variety of search-free methods have been studied. An excellent review and comparison of these existing techniques are provided in [5], [6]. The manifold separation technique (MST) is utilized to extend the conventional root-MUSIC to arbitrary arrays in [7]–[10]. Nevertheless, only azimuth estimation is considered in the above works. In [11], joint azimuth and elevation estimation for arbitrary arrays is concerned, in which the MST model is used to convert the 2D DOA estimation to a problem of bivariate polynomial rooting. However, if the array aperture becomes large, the order of the bivariate polynomial must be quite large which leads to unacceptable computational complexity for the rooting procedure.

In this paper we build on our previous works of [12], [13] to propose a computationally attractive 2D DOA estimator for arbitrary arrays. First, we convert the 2D-MUSIC cost function into standard two-dimensional discrete Fourier transform (DFT) form by using the MST. In doing so, the 2D spatial spectrum can be efficiently computed by using the 2D fast Fourier transform (FFT) algorithms. For the sake of real-time implementation, we present a highly parallel FFT-based algorithm to compute the 2D spatial spectrum. Such method not only alleviates the computation burden of root-MUSIC or MUSIC; it is also easy for hardware implementation. Moreover, in order to improve the estimation performance, we propose a multidimensional-subspace-fitting method by using the first-order Taylor expansion.

The rest of this paper is organized as follows. In Section II, the model for 2D wavefield MST is recalled. We present our FFT-based algorithm for computing the 2D spatial spectrum in Section III, which consists of two main stages. Then, by using the first-order Taylor expansion, a multidimensional-subspace-fitting approach is formulated to improve the estimation performance in Section IV. The simulation results are shown and analyzed in Section V. Finally, the paper is concluded in Section VI.

II. 2D Wavefield Manifold Separation Technique

By using the Fourier series expansion, the periodic array steering vector can be decomposed as

$$\mathbf{a}(\theta,\phi) = \mathbf{G}\mathbf{v}(\theta,\phi) + \boldsymbol{\varepsilon} \in \mathcal{C}^{L\times 1} \tag{7}$$

where $\boldsymbol{\varepsilon}$ denotes the modeling error. This is the so-called manifold separation technique [7], [9], [14]. The vector $\mathbf{v}(\theta,\phi)$ is the spatial Fourier basis consisting of the following Vandermonde-structured vectors

$$\mathbf{v}(\theta,\phi) = \mathbf{v}(\theta)\otimes\mathbf{v}(\phi) \in \mathcal{C}^{M^2\times 1} \tag{8}$$

where

$$\mathbf{v}(\theta) = \left[z_\theta^{\frac{M-1}{2}},\ldots,1,\cdots,z_\theta^{-\frac{M-1}{2}}\right]^T \in \mathcal{C}^{M\times 1},$$
$$\mathbf{v}(\phi) = \left[z_\phi^{\frac{M-1}{2}},\ldots,1,\ldots,z_\phi^{-\frac{M-1}{2}}\right]^T \in \mathcal{C}^{M\times 1} \tag{9}$$

with $z_\theta = e^{j\theta}$, $z_\phi = e^{j\phi}$ and $\otimes$ representing the Kronecker product. We can see that the vector $\mathbf{v}(\theta,\phi)$ depends only on the wavefield. The model error $\boldsymbol{\varepsilon}$ can be safely neglected when the mode number M is large enough. Note that different mode numbers can be chosen for $\mathbf{v}(\theta)$ and $\mathbf{v}(\phi)$. Without loss of generality, we choose the same mode number for both azimuth and elevation vectors in this paper. Additionally, the matrix $\mathbf{G}\in\mathcal{C}^{L\times M^2}$ is called the sampling matrix which depends only on the sensor array configuration and its properties. It is clear that the essence of the MST model in (7) is using a number of 2D Fourier series to approximate the array response.

We can acquire the sampling matrix through measurements from a number of different location of angles. Typically, these calibration measurements are obtained in controlled environments such as anechoic chambers. The antenna array is mounted on a mechanical platform and then we rotate it about its centroid, while a known active source is held fixed. This process creates a discrete set of measured points. Let $\boldsymbol{\mathcal{A}}_\ell(\theta_c,\phi_c)\in\mathcal{C}^{Q_e\times Q_a}$ denote the calibration measurement matrix for the ℓ-th antenna where Q_a and Q_e, respectively, represent the numbers of calibration points along $\theta\in[0,2\pi]$ and $\phi\in[0,\pi]$. In order to make the measurement data periodic in elevation with period 2π, we construct a matrix $\boldsymbol{\mathcal{A}}_\ell^r(\theta_c,\phi_c)\in\mathcal{C}^{(Q_e-2)\times Q_a}$ from the measured $\boldsymbol{\mathcal{A}}_\ell(\theta_c,\phi_c)$ by a shift of 180° in azimuth and a flip in elevation followed by discarding the first and last row to avoid the redundancy at the poles (i.e., 0° and 180° elevations) [11]. We can combine these two matrices into a matrix

$$\boldsymbol{\mathcal{A}}'_\ell = \begin{bmatrix}\boldsymbol{\mathcal{A}}_\ell(\theta_c,\phi_c)\\ \boldsymbol{\mathcal{A}}_\ell^r(\theta_c,\phi_c)\end{bmatrix} \in \mathcal{C}^{(2Q_e-2)\times Q_a}. \tag{10}$$

The sampling matrix $\mathbf{G}_\ell\in\mathcal{C}^{(2Q_e-2)\times Q_a}$ for the ℓ-th sensor corresponds to the 2D inverse discrete Fourier transform of $\boldsymbol{\mathcal{A}}'_\ell$. Then stacking all the rows $\{\mathrm{vec}\{\mathbf{G}_\ell\}^T\}_{\ell=1}^L$ (where $\mathrm{vec}\{\cdot\}$ denotes the vectorization operator) and keeping the middle M^2 columns, we can obtain the sampling matrix

$$\mathbf{G} = \begin{bmatrix}\mathrm{vec}\{\mathbf{G}_1\}^T\\ \vdots\\ \mathrm{vec}\{\mathbf{G}_L\}^T\end{bmatrix} \in \mathcal{C}^{L\times M^2}. \tag{11}$$

For more details, see [11], [14], [15]. In this paper, we assume that the sampling matrix has been obtained off-line prior to DOA estimations and we place focus on the DOA estimations.

The aim of MST is twofold. First, we can exploit the Vandermonde structure in (7) to apply computationally efficient DOA estimators to arbitrary arrays. Second, the sampling matrix can accommodate the array nonidealities, e.g., mutual coupling, physical location misplacement of array elements and mounting

platform reflections, implying that the model in (7) is suitable for real-world arrays.

III. FFT-Based 2D MUSIC

This section provides an FFT-based method to compute the 2D spatial spectrum with arbitrary array configurations. Inserting the MST expansion model of (7) into the MUSIC cost function in (6), the spatial spectrum becomes

$$\begin{aligned} p(\theta,\phi) &= \mathbf{v}(\theta,\phi)^H \left(\mathbf{G}^H \widehat{\mathbf{E}}_n \widehat{\mathbf{E}}_n^H \mathbf{G}\right) \mathbf{v}(\theta,\phi) \\ &= \mathbf{p}(\theta)^T \mathbf{C} \mathbf{p}(\phi) \\ &= \sum_{k_1=-(M-1)}^{M-1} \sum_{k_2=-(M-1)}^{M-1} [\mathbf{C}]_{M+k_1,M+k_2} z_\theta^{-k_1} z_\phi^{-k_2} \end{aligned} \tag{12}$$

where $\mathbf{p}(\theta)=[z_\theta^{M-1},\ldots,0,\ldots,z_\theta^{-(M-1)}]^T$ and $\mathbf{p}(\phi)=[z_\phi^{M-1},\ldots,0,\ldots,z_\phi^{-(M-1)}]^T$ are Vandermonde structured vectors. The notation $[\cdot]_{m,n}$ stands for the (m,n)-th element of the matrix between square brackets. The matrix $\mathbf{C}\in\mathcal{C}^{(2M-1)\times(2M-1)}$ is called the coefficient matrix. Apparently, the expression in (12) is of typical 2D DFT form and therefore we can apply the well-known 2D FFT algorithm to calculate the 2D spatial spectrum efficiently. Next, we will show how to efficiently compute the the coefficient matrix and the 2D spatial spectrum.

A. Coefficient Matrix Computation

The coefficient matrix $\mathbf{C}$ can be found by the following two steps [11]. First, we express the matrix $\mathbf{B}\triangleq\mathbf{G}^H\widehat{\mathbf{E}}_n\widehat{\mathbf{E}}_n^H\mathbf{G}$ in block form as

$$\mathbf{B}=\begin{bmatrix} \mathbf{B}_{1,1} & \mathbf{B}_{1,2} & \cdots & \mathbf{B}_{1,M} \\ \mathbf{B}_{2,1} & \mathbf{B}_{2,2} & \cdots & \mathbf{B}_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_{M,1} & \mathbf{B}_{M,2} & \cdots & \mathbf{B}_{M,M} \end{bmatrix} \tag{13}$$

where each sub-block matrix is an M-by-M matrix. Then computing the sum of the block elements along all $2M-1$ diagonals produces

$$\mathbf{D}=\begin{bmatrix}\mathbf{D}_1 & \mathbf{D}_2 & \ldots & \mathbf{D}_{2M-1}\end{bmatrix} \tag{14}$$

with

$$\mathbf{D}_i=\sum_{\forall M-(m-n)=i}\mathbf{B}_{m,n}. \tag{15}$$

Secondly, for the (i,q)-th element of $\mathbf{C}$, we compute the sum of the elements along the q-th diagonal of $\mathbf{D}_i$ to produce

$$[\mathbf{C}]_{i,q}=\sum_{\forall M-(m-n)=q}[\mathbf{D}_i]_{m,n} \tag{16}$$

where $1\le i,q\le 2M-1$.

Alternatively, here we present a faster one-step operation to compute the coefficients. The sample matrix can be rewritten in block form as

$$\mathbf{G}=\begin{bmatrix}\mathbf{G}_1,\mathbf{G}_2,\ldots\mathbf{G}_M\end{bmatrix} \tag{17}$$

where the size of each sub-block matrix is L-by-M. Taking the summations along the i-th diagonal of $\mathbf{G}^H\widehat{\mathbf{E}}_n\widehat{\mathbf{E}}_n^H\mathbf{G}$, we obtain

$$\begin{aligned} \mathbf{D}_i &= \sum_{\forall M-(m-n)=i}\mathbf{G}_m^H\widehat{\mathbf{E}}_n\widehat{\mathbf{E}}_n^H\mathbf{G}_n \\ &= \sum_{k=K+1}^{L}\sum_{\forall M-(m-n)=i}\mathbf{G}_m^H\mathbf{e}_k\mathbf{e}_k^H\mathbf{G}_n \\ &= \sum_{k=K+1}^{L}\sum_{\forall M-(m-n)=i}(\mathbf{G}_m^H\mathbf{e}_k)(\mathbf{G}_n^H\mathbf{e}_k)^H \end{aligned} \tag{18}$$

where the noise eigenvectors $\widehat{\mathbf{E}}_n=[\mathbf{e}_{K+1},\ldots,\mathbf{e}_L]$.

Inspired from [16], the sum of the entries of the q-th diagonal of any outer product $\mathbf{x}\mathbf{y}^H$ (where $\mathbf{x},\mathbf{y}\in\mathcal{C}^{L\times 1}$) can be accomplished by

$$\sum_{\forall L-(m-n)=q}[\mathbf{x}\mathbf{y}^H]_{m,n}=\mathbf{y}^H\mathbf{J}_{L,q}\mathbf{x} \tag{19}$$

where the shift matrix $\mathbf{J}_{L,q}\in\mathcal{R}^{L\times L}$ is defined as

$$\mathbf{J}_{L,q}=\begin{cases}\begin{bmatrix}\mathbf{O} & \mathbf{I}_q\\ \mathbf{O} & \mathbf{O}\end{bmatrix}, & \text{if } 1\le q\le L \\ \begin{bmatrix}\mathbf{O} & \mathbf{O}\\ \mathbf{I}_{2L-q} & \mathbf{O}\end{bmatrix}, & \text{if } L+1<q\le 2L-1\end{cases} \tag{20}$$

with the notation $\mathbf{O}$ denoting the zero matrix of appropriate size. As a consequence of this property, the sum of the elements along the q-th diagonal of $\mathbf{D}_i$ can be obtained by

$$\begin{aligned} [\mathbf{C}]_{i,q} &= \sum_{\forall M-(m-n)=q}[\mathbf{D}_i]_{m,n} \\ &= \sum_{k=K+1}^{L}\sum_{\forall M-(m-n)=i}\mathbf{e}_k^H\mathbf{G}_n\mathbf{J}_{M,q}\mathbf{G}_m^H\mathbf{e}_k \\ &= \sum_{k=K+1}^{L}\mathbf{e}_k^H\widetilde{\mathbf{G}}_{i,q}\mathbf{e}_k \\ &= \mathrm{Tr}\left\{\widetilde{\mathbf{G}}_{i,q}\mathbf{\Pi}_n\right\} \end{aligned} \tag{21}$$

where $\mathbf{\Pi}_n=\widehat{\mathbf{E}}_n\widehat{\mathbf{E}}_n^H=\mathbf{I}_L-\widehat{\mathbf{E}}_s\widehat{\mathbf{E}}_s^H$ denotes the noise-subspace projector, $\mathrm{Tr}\{\cdot\}$ denotes the trace of the matrix between braces and the $L\times L$ matrix

$$\widetilde{\mathbf{G}}_{i,q}\triangleq\sum_{\forall M-(m-n)=i}\mathbf{G}_n\mathbf{J}_{M,q}\mathbf{G}_m^H \tag{22}$$

is independent of any signal and therefore can be formed off-line. Here we assume that $K<L-K$ and therefore we use the signal eigenvectors instead of the noise eigenvectors to reduce the computational complexity.

In practice, the magnitude of the matrix $\mathbf{C}$ is concentrated within a small support area centered at $[\mathbf{C}]_{M,M}$. Therefore, instead of the full coefficient matrix, a truncated version is used for further processing. More specifically, only the $2M_t+1$ middle rows and middle columns of $\mathbf{C}$ are needed (where $M_t<M$),

which is defined by MATLAB notation as

$$\mathbf{C}_t = \mathbf{C}(M - M_t : M + M_t, M - M_t : M + M_t). \quad (23)$$

It is difficult to find an analytical expression for M_t, but a convenient rule of thumb $M_t \approx 2\kappa R$ has been suggested in [17], where κ is the signal wavenumber and R is the largest distance between the array sensors and the origin of the coordination system. We can see that such value marginally meets the Nyquist criterion. In practical application, oversampling is often used to increase the resolution of spatial estimation. In our simulation, we find that

$$M_t \approx 2.5\kappa R \quad (24)$$

is enough. Moreover, we have the following symmetry properties

$$\begin{aligned} [\mathbf{C}]_{M-k_1,M-k_2} &= ([\mathbf{C}]_{M+k_1,M+k_2})^* \\ [\mathbf{C}]_{M+k_1,M-k_2} &= (-1)^{k_1}[\mathbf{C}]_{M+k_1,M+k_2} \\ [\mathbf{C}]_{M-k_1,M+k_2} &= (-1)^{k_1}([\mathbf{C}]_{M+k_1,M+k_2})^* \end{aligned} \quad (25)$$

where $(\cdot)^*$ denotes the complex conjugate. The proof of the above can be found in Appendix A. This property suggests that computing a quarter coefficients (i.e., $[\mathbf{C}]_{M+k_1,M+k_2}$ for $0 \le k_1, k_2 \le M_t$) is sufficient to find the coefficients of interest.

B. Partial 2D DFT

Typically, 2D DFTs are implemented in a row-column (RC) decomposition manner. For instance, the column-wise 1D DFTs are executed for all columns and then followed by the row-wise 1D DFTs for all rows. However, such traditional manner may cause high latency for our case. This is because the DFT point number, denoted by N, is required to be relatively large for the purpose of reducing grid error. Next, we will introduce a partial 2D DFT algorithm which is highly parallel and requires a small number of memory accesses. Different than the partial DFT presented in [18], [19] (where irregularly-structured zoom-FFT is used), our proposed method is regular and scalable.

Let us pad the $(2M_t + 1) \times (2M_t + 1)$ coefficient matrix $\mathbf{C}_t$ with zeros to create a $r \times r$ matrix $\mathbf{C}_r$ where the number r, no less than $(2M_t + 1)$, takes the value of a power of 2. This means $r = 2^{\lceil \log_2(2M_t+1) \rceil}$ where $\lceil \cdot \rceil$ represents the ceiling function. Then we construct an $N \times N$ coefficient matrix $\mathbf{C}_N$

$$\mathbf{C}_N = \begin{bmatrix} \mathbf{C}_r & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix}_{N \times N} \quad (26)$$

where only the top-left corner contains non-zero elements. Following the work of [20], the 2D DFT in (12) can be represented in matrix form as follows:

$$\mathbf{C}_{\text{DFT}} = \mathbf{F}_N \mathbf{C}_N \mathbf{F}_N^T \quad (27)$$

where $(\cdot)^T$ denotes the transpose operation and $\mathbf{F}_N$ stands for the $N \times N$ Fourier transform matrix whose (m, n)-th entry is $[\mathbf{F}_N]_{m,n} = \omega_N^{mn}$ with $\omega_N = e^{-j\frac{2\pi}{N}}$. Moreover, we can factorize the Fourier transform matrix into a product of sparse matrices as follows [21]

$$\mathbf{F}_N = (\mathbf{F}_r \otimes \mathbf{I}_p)\, \mathbf{T}_{N,p}\, (\mathbf{I}_r \otimes \mathbf{F}_p)\, \mathbf{P}_{N,r} \quad (28)$$

where $p = N/r$. The matrix $\mathbf{T}_{N,p}$ is an $N \times N$ diagonal matrix of twiddle factors whose (n, n)-th entry is given by

$$[\mathbf{T}_{N,p}]_{n,n} = \omega_N^{\left((n-1 \bmod p)\lfloor \frac{n-1}{p} \rfloor\right)} \quad (29)$$

where mod denotes the modulus operation and $\lfloor \cdot \rfloor$ represents the floor function. The matrix $\mathbf{P}_{N,r}$ is the stride-by-r permutation matrix whose (m, n)-th entry is given by

$$[\mathbf{P}_{N,r}]_{m,n} = \begin{cases} 1, & \text{iff } n - 1 = (m-1)r \bmod N + \lfloor \frac{(m-1)r}{N} \rfloor \\ 0, & \text{otherwise.} \end{cases} \quad (30)$$

Note that multiplying by the stride permutation matrix does not cause any computation complexity since only the column or row order is rearranged.

Now inserting (28) into (27), we obtain (35). By exploiting the fact that only $r \times r$ elements at the top-left corner of $\mathbf{C}_N$ are non-zero, the matrix $\mathbf{C}_1$ defined in (35) can be obtained without any multiplication or addition operations. More specifically, we simply duplicate the elements of $\mathbf{C}_r$ to obtain $\mathbf{C}_1$ as follows

$$\mathbf{C}_1 = \begin{bmatrix} c_{1,1}\mathbf{1}_{p\times p} & c_{1,2}\mathbf{1}_{p\times p} & \cdots & c_{1,r}\mathbf{1}_{p\times p} \\ c_{2,1}\mathbf{1}_{p\times p} & c_{2,2}\mathbf{1}_{p\times p} & \cdots & c_{2,r}\mathbf{1}_{p\times p} \\ \vdots & \vdots & \ddots & 0 \\ c_{r,1}\mathbf{1}_{p\times p} & c_{r,2}\mathbf{1}_{p\times p} & \cdots & c_{r,r}\mathbf{1}_{p\times p} \end{bmatrix}_{N\times N} \quad (31)$$

where $c_{m,n} = [\mathbf{C}_r]_{m,n}$ and $\mathbf{1}_{p\times p}$ is a $p \times p$ matrix of ones. In addition, we have

$$\begin{aligned} \mathbf{C}_2 &\triangleq \mathbf{T}_{N,p}\mathbf{C}_1\mathbf{T}_{N,p}^T \\ &= \mathbf{C}_1 \odot \widetilde{\mathbf{T}}_{N,p} \end{aligned} \quad (32)$$

where $\odot$ is the Schur-Hadamard product and the (m, n)-th entry of $\widetilde{\mathbf{T}}_{N,p}$ is given by

$$[\widetilde{\mathbf{T}}_{N,p}]_{m,n} = [\mathbf{T}_{N,p}]_{m,m}[\mathbf{T}_{N,p}]_{n,n}. \quad (33)$$

Note that the matrix $\widetilde{\mathbf{T}}_{N,p}$ is no longer diagonal and it can be found in a "once-and-for-all" fashion. Thus the 2D DFT in (35) becomes

$$\begin{aligned} \mathbf{C}_{\text{DFT}} &= (\mathbf{F}_r \otimes \mathbf{I}_p)\, \mathbf{C}_2\, (\mathbf{F}_r \otimes \mathbf{I}_p) \\ &= \underbrace{\mathbf{P}_{N,r}\, (\mathbf{I}_p \otimes \mathbf{F}_r)\, \underbrace{\mathbf{P}_{N,p}\mathbf{C}_2\mathbf{P}_{N,r}}_{\triangleq \mathbf{C}_3}\, (\mathbf{I}_p \otimes \mathbf{F}_r)\, \mathbf{P}_{N,p}}_{\triangleq \mathbf{C}_4} \end{aligned} \quad (34)$$

where the property $\mathbf{F}_r \otimes \mathbf{I}_p = \mathbf{P}_{N,r}\, (\mathbf{I}_p \otimes \mathbf{F}_r)\, \mathbf{P}_{N,p}$ is utilized in the above. To obtain the matrix $\mathbf{C}_4 \triangleq (\mathbf{I}_p \otimes \mathbf{F}_r)\, \mathbf{C}_3\, (\mathbf{I}_p \otimes \mathbf{F}_r)$, we first divide the matrix $\mathbf{C}_3$ into $p \times p$ subblocks each of which is a $r \times r$ matrix. Then we perform r-point 2D DFT for each subblock. It is worth pointing out that such 2D DFTs on all subblocks can be executed in parallel. Lastly, we permute the matrix $\mathbf{C}_4$ to produce the final 2D spatial spectrum. (35) shown at the bottom of the next page.

Algorithm 1: FFT-based 2D MUSIC.

- *Off-line process*:
  1) For the calibration purpose, measure the array response at a number of locations and calculate the sampling matrix $\mathbf{G}$;
  2) Compute $(M_t+1)^2$ matrices $\widetilde{\mathbf{G}}_{i,q}$ defined in (22) for all $i,q \in [M, M+M_t]$;
  3) Compute the matrix $\widetilde{\mathbf{T}}_{N,p}$ in (33);
- *Online process*:
  1) Estimate the covariance matrix $\widehat{\mathbf{R}}$ via (4) and perform eigen-decomposition to obtain the signal eigenvector matrix $\widehat{\mathbf{E}}_s$;
  2) Compute a quarter coefficients of the truncated matrix $\mathbf{C}_t$ by (21) in parallel;
  3) Apply partial 2D DFT operation:
     a) Pad $\mathbf{C}_t$ with zeros to create $\mathbf{C}_r$ if needed and then duplicate $\mathbf{C}_r$ to create $\mathbf{C}_1$ as (31);
     b) Compute the matrix $\mathbf{C}_2$ by (32) and then permute it to produce $\mathbf{C}_3$;
     c) Perform r-point 2D FFT on $p \times p$ sub-blocks of $\mathbf{C}_3$ simultaneously to yield $\mathbf{C}_4$ and then permute $\mathbf{C}_4$ to obtain the final 2D spatial spectrum.

C. Summary of FFT-Based 2D MUSIC

In summary, we propose Algorithm 1 to compute the 2D spatial spectrum. Note that the off-line process needs to be done only once for a given antenna array.

Here we analyze the computational complexity of Algorithm 1. Note that we mainly consider the multiplication operations which results in major computation in the estimators. The complexity to find the K-dimension $\widehat{\mathbf{E}}_s$, using the fast subspace decompositon technique [22], is given by $O(KL^2)$. For each element of the matrix $\mathbf{C}_t$, $O(L^2)$ complex multiplications are needed [see (21)]. Thus the complexity to compute the truncated matrix $\mathbf{C}_t$ is $O(M_t^2L^2)$. For Step 3.b, the complexity is $O(N^2)$. In addition, the complexity of Step 3.c is $O(p \times p \times (r^2 \log r)) = O(N^2 \log r)$ in which each r-point FFT requires $O(\frac{r}{2}\log r)$ complex multiplications. Therefore, the overall complexity of the proposed algorithm is $O(KL^2 + M_t^2L^2 + N^2 + N^2 \log r)$. In comparison, the complexity of MUSIC is $O(KL^2 + N_aN_eKL)$ [5], [23] where N_a and N_e represent the numbers of searched spectral points along the azimuth and elevation respectively. In order to achieve high spectral resolution (or reduce grid error), we have to choose a large number for the searched spectral points [23]. As a result, computing spectral points dominates the complexity of MUSIC methods.

In Table I, we compare the traditional 2D MUSIC with Algorithm 1 in terms of the computational complexity. Note that the common term $O(KL^2)$ (required by the eigen-decomposition) is not considered here. The parameter $M_t \approx 2.5\kappa R$ is used for the truncated coefficient matrix. The array aperture is assumed to be $(L-1)$ half-wavelengths, and thus the array radius is $R = (L-1)/4$ in units of wavelengths. The signal number $K = 3$ is taken. The search point numbers are $N = N_a = 4096$ and $N_e = 2048$ (since we care the elevation $\phi \in [0, \pi]$ only). The comparison is also illustrated in Fig. 1. We can see that the impact on the term $O(N^2 + N^2 \log r)$ due to the antenna increase is quite moderate. Despite of the fact that the term $O(M_t^2L^2)$ increases quadratically with the antenna number L and becomes significant when L is over 30, the overall complexity of our method is much less than that of the traditional MUSIC. It is worthwhile to point out that the complexity of our method is almost unaffected by the signal number whereas that of the traditional MUSIC increases linearly with the signal number.

Before closing this section, we would like to point out the main advantages of our proposed algorithm: 1) the complexity to compute the search points appears almost unaffected by the signal number; 2) our algorithm is highly parallel and therefore we can reduce further the complexities of Step 2 and 3.c of the online process to as little as $O(L^2)$ and $O(r^2 \log r)$ using multiple simple processors; 3) our method can be implemented on embedded systems where resources and memory are limited because the sizes of matrix operations (including the coefficient computation and partial 2D DFT) are relatively small.

IV. Multidimensional Subspace Fitting Using the First-Order Taylor Expansion

A. Initial Angle Estimates

By searching the K minima of the 2D MUSIC spectrum, we can obtain an initial estimate of the azimuth and elevation angles. When the signals are closely spaced, however, the MUSIC methods may fail to distinguish them. Assume that K_0 estimates are found by searching the MUSIC spatial spectrum where $K_0 \le K$. In order to find the remaining DOAs, we employ the iterative method presented in [24]. At each iteration, we find the angular estimates of a single source by maximizing

$$(\widehat{\theta}_{k+1}, \widehat{\phi}_{k+1}) = \arg\max_{\theta,\phi} \mathrm{Tr}\left\{\mathbf{\Pi}_{[\mathbf{A}_k,\, \mathbf{a}(\theta,\phi)]}\widehat{\mathbf{R}}\right\} \tag{36}$$

where $\mathbf{A}_k$ represents the available signal steering vectors before iteration. The notation $\mathbf{\Pi}_{\mathbf{A}}$ stands for the projection operator onto the subspace spanned by $\mathbf{A}$, which can be given by

$$\mathbf{\Pi}_{\mathbf{A}} = \mathbf{A}\mathbf{A}^{\dagger} \tag{37}$$

$$\mathbf{C}_{\mathrm{DFT}} = (\mathbf{F}_r \otimes \mathbf{I}_p)\,\mathbf{T}_{N,p}\underbrace{\underbrace{(\mathbf{I}_r \otimes \mathbf{F}_p)\,\mathbf{P}_{N,r}\mathbf{C}_N\mathbf{P}_{N,r}^T\,(\mathbf{I}_r \otimes \mathbf{F}_p)}_{\triangleq \mathbf{C}_1}\,\mathbf{T}_{N,p}^T}_{\triangleq \mathbf{C}_2}\,(\mathbf{F}_r \otimes \mathbf{I}_p) \tag{35}$$

TABLE I
COMPARISONS OF COMPUTATIONAL COMPLEXITY

Antenna Number		$L=10$	$L=20$	$L=30$	$L=40$	$L=50$
Traditional MUSIC		2.5×10^8	5.0×10^8	7.5×10^8	1.0×10^9	1.3×10^9
Proposed	$O(M_t^2L^2)$	1.3×10^5	2.3×10^6	1.2×10^7	3.8×10^7	9.3×10^7
	$O(N^2+N^2\log r)$	1.3×10^8	1.5×10^8	1.5×10^8	1.6×10^8	1.7×10^8
	Total	1.3×10^8	1.5×10^8	1.6×10^8	2.0×10^8	2.6×10^8

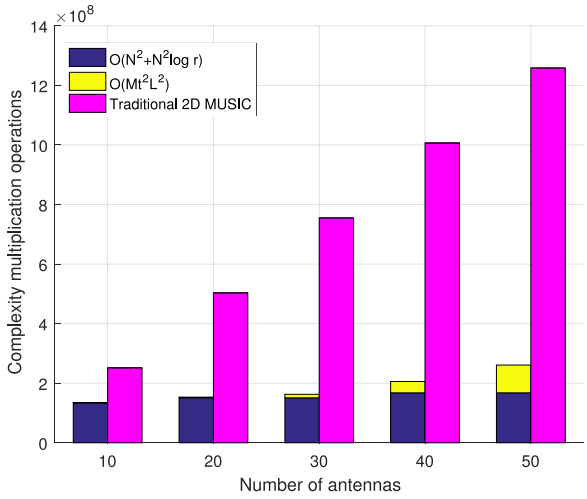


Fig. 1. Comparison of computational complexities where the proposed Algorithm 1 includes two terms: $O(M_t^2L^2)$ and $O(N^2+N^2\log r)$.

where the notation $\mathbf{A}^\dagger=(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$ denotes the Moore-Penrose pseudo-inverse. In addition, the projection matrix to the orthogonal complement of $\mathbf{A}$ is given by

$$\mathbf{\Pi}_{\mathbf{A}}^{\perp}=\mathbf{I}_L-\mathbf{\Pi}_{\mathbf{A}}. \tag{38}$$

The projection matrix $\mathbf{\Pi}_{[\mathbf{A}_k,\,\mathbf{a}(\theta,\phi)]}$ can be decomposed as [24]

$$\mathbf{\Pi}_{[\mathbf{A}_k,\,\mathbf{a}(\theta,\phi)]}=\mathbf{\Pi}_{\mathbf{A}_k}+\mathbf{\Pi}_{\left(\mathbf{\Pi}_{\mathbf{A}_k}^{\perp}\mathbf{a}(\theta,\phi)\right)}. \tag{39}$$

Applying the above to (36), we have

$$(\widehat{\theta}_{k+1},\widehat{\phi}_{k+1})=\arg\max_{\theta,\phi}\frac{\mathbf{a}^H(\theta,\phi)\mathbf{\Pi}_{\mathbf{A}_k}^{\perp}\widehat{\mathbf{R}}\mathbf{\Pi}_{\mathbf{A}_k}^{\perp}\mathbf{a}(\theta,\phi)}{\mathbf{a}^H(\theta,\phi)\mathbf{\Pi}_{\mathbf{A}_k}^{\perp}\mathbf{a}(\theta,\phi)}. \tag{40}$$

Obviously, we can conduct the previously proposed FFT-based algorithm twice to compute the numerator and denominator of (40) for different azimuths and elevations.

B. Multidimensional Subspace Fitting

For notational simplicity, we use $\mathbf{\Pi}_{\boldsymbol{\eta}}$ to represent $\mathbf{\Pi}_{\mathbf{A}(\boldsymbol{\eta})}$ where $\boldsymbol{\eta}=[\theta_1,\ldots,\theta_K,\phi_1,\ldots,\phi_K]^T$ and

$$\mathbf{A}(\boldsymbol{\eta})=\mathbf{G}[\mathbf{v}(\theta_1)\otimes\mathbf{v}(\phi_1),\ldots\mathbf{v}(\theta_K)\otimes\mathbf{v}(\phi_K)]. \tag{41}$$

Now let us consider the following multidimensional subspace fitting problem

$$\min_{\boldsymbol{\eta}} f\triangleq\min_{\boldsymbol{\eta}}\mathrm{Tr}\left\{\mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\mathbf{\Psi}\right\} \tag{42}$$

where $\mathbf{\Psi}=\widehat{\mathbf{R}}$ gives the maximum likelihood estimator (MLE) [24] and $\mathbf{\Psi}=\widehat{\mathbf{E}}_s(\mathbf{\Lambda}_s-\sigma_n^2\mathbf{I}_K)^2\widehat{\mathbf{\Lambda}}_s^{-1}\widehat{\mathbf{E}}_s^H$ gives the weighted subspace fitting algorithm [26], [27]. In [11], [27], a Modified Variable Projection (MVP) method has been presented to solve (42) by iterating

$$\boldsymbol{\eta}^{(i+1)}=\boldsymbol{\eta}^{(i)}-\mu_i(\nabla^2 f)^{-1}\nabla f \tag{43}$$

where $\mu_i<1\in\mathcal{R}$ represents the stepsize, $\nabla^2 f\in\mathcal{R}^{2K\times 2K}$ and $\nabla f\in\mathcal{R}^{2K\times 1}$, respectively, stand for the Hessian matrix and gradient of the cost function evaluated in $\boldsymbol{\eta}^{(i)}$. However, it is difficult to always properly choose the stepsize for such Newton-type iteration. Moveover, in each iteration, both the Hessian matrix and gradient have to be updated which increases the computational complexity. Next, we present a more convenient method in which the problems of choosing stepsize and computing the Hessian matrix are both circumvented.

Applying the first-order Taylor expansion to the projection operator can result in [25]

$$\mathbf{\Pi}_{(\boldsymbol{\eta}_0+\Delta\boldsymbol{\eta})}\approx\mathbf{\Pi}_{\boldsymbol{\eta}_0}+\sum_{k=1}^{2K}\Delta\eta_k\left.\frac{\partial\mathbf{\Pi}_{\boldsymbol{\eta}}}{\partial\eta_k}\right|_{\boldsymbol{\eta}_0} \tag{44}$$

where the partial derivatives of the projector are provided in Appendix B. If we insert the Taylor expansion of (44) into (42) straightforwardly, we cannot find the solution for the vector $\Delta\boldsymbol{\eta}$. The trick in this paper is that we use the idempotent property of projection operator to transform (42) equivalently to

$$\min_{\boldsymbol{\eta}} f=\min_{\boldsymbol{\eta}}\mathrm{Tr}\left\{\left(\mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\right)^2\mathbf{\Psi}\right\}. \tag{45}$$

Then differentiating the above objective function with respect to the i-th ($i=1,2,\ldots,2K$) element of $\Delta\boldsymbol{\eta}$, we have

$$\begin{aligned}\frac{\partial f}{\partial\Delta\eta_i}&=\mathrm{Tr}\left\{\left(\mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\mathbf{\Psi}+\mathbf{\Psi}\mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\right)\frac{\partial\mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}}{\partial\Delta\eta_i}\right\}\\&=-\mathrm{Tr}\left\{\left(2\mathbf{\Psi}-\mathbf{\Pi}_{\boldsymbol{\eta}}\mathbf{\Psi}-\mathbf{\Psi}\mathbf{\Pi}_{\boldsymbol{\eta}}\right)\frac{\partial\mathbf{\Pi}_{\boldsymbol{\eta}}}{\partial\Delta\eta_i}\right\}\end{aligned} \tag{46}$$

Algorithm 2: Proposed Multidimensional-Subspace-Fitting Estimator.

1) Employ Algorithm 1 and search the 2D spatial spectrum to obtain an initial estimate $\boldsymbol{\eta}_0 = [\widehat{\theta}_1, \ldots, \widehat{\theta}_{K_0}, \widehat{\phi}_1, \ldots, \widehat{\phi}_{K_0}]^T$,
2) Compute the error vector $\Delta\boldsymbol{\eta}$ using (52) where $\mathbf{H}$ and $\mathbf{r}$ are defined in (50) and (51),
3) Update $\boldsymbol{\eta}_0 = \boldsymbol{\eta}_0 + \Delta\boldsymbol{\eta}$,
4) If $K_0 < K$, for $k = K_0$ to $K-1$, do the follows
   a) Form $\mathbf{A}_k$ by all the available signal steering vectors and compute the orthogonal projector $\boldsymbol{\Pi}^{\perp}_{\mathbf{A}_k}$,
   b) Compute the numerator and denominator of (40) using the FFT-based method described in Section III,
   c) Compute the fractions and search $(\widehat{\theta}_{k+1}, \widehat{\phi}_{k+1})$ corresponding to the maximal fraction.
   d) Add $(\widehat{\theta}_{k+1}, \widehat{\phi}_{k+1})$ to the vector $\boldsymbol{\eta}_0$ and set $K_0 = K_0 + 1$.
5) Repeat Step 2, 3 and 4 a few times.

where $\frac{\partial \boldsymbol{\Pi}^{\perp}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i} = -\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i}$ is used in the above. Setting (46) to zero, we obtain

$$\mathrm{Tr}\left\{ \left(\boldsymbol{\Pi}_{\boldsymbol{\eta}}\boldsymbol{\Psi} + \boldsymbol{\Psi}\boldsymbol{\Pi}_{\boldsymbol{\eta}}\right) \frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i} \right\} = \mathrm{Tr}\left\{ 2\boldsymbol{\Psi} \frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i} \right\}. \tag{47}$$

Then replacing $\boldsymbol{\Pi}_{\boldsymbol{\eta}}$ with (44) and after some algebraic manipulations, we get

$$\sum_{k=1}^{2K} \mathrm{Tr}\left\{ \left(\boldsymbol{\Psi} \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_k}\right|_{\boldsymbol{\eta}_0} + \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_k}\right|_{\boldsymbol{\eta}_0} \boldsymbol{\Psi} \right) \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i}\right|_{\boldsymbol{\eta}_0} \right\} \Delta\eta_k$$
$$= \mathrm{Tr}\left\{ \left(2\boldsymbol{\Psi} - \boldsymbol{\Pi}_{\boldsymbol{\eta}_0}\boldsymbol{\Psi} - \boldsymbol{\Psi}\boldsymbol{\Pi}_{\boldsymbol{\eta}_0}\right) \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i}\right|_{\boldsymbol{\eta}_0} \right\}. \tag{48}$$

Moreover, the equations $[\frac{\partial f}{\partial \Delta\eta_1}, \ldots, \frac{\partial f}{\partial \Delta\eta_{2K}}]^T = \mathbf{0}^T$ can be written more compactly as

$$\mathbf{H}\Delta\boldsymbol{\eta} = \mathbf{r} \tag{49}$$

where

$$[\mathbf{H}]_{i,k} = \mathrm{Tr}\left\{ \left(\boldsymbol{\Psi} \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_k}\right|_{\boldsymbol{\eta}_0} + \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_k}\right|_{\boldsymbol{\eta}_0} \boldsymbol{\Psi} \right) \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i}\right|_{\boldsymbol{\eta}_0} \right\} \tag{50}$$

and

$$[\mathbf{r}]_i = \mathrm{Tr}\left\{ \left(2\boldsymbol{\Psi} - \boldsymbol{\Pi}_{\boldsymbol{\eta}_0}\boldsymbol{\Psi} - \boldsymbol{\Psi}\boldsymbol{\Pi}_{\boldsymbol{\eta}_0}\right) \left.\frac{\partial \boldsymbol{\Pi}_{\boldsymbol{\eta}}}{\partial \Delta\eta_i}\right|_{\boldsymbol{\eta}_0} \right\} \tag{51}$$

with $i, k = 1, 2, \ldots, 2K$. Now the error vector can be readily computed by

$$\Delta\boldsymbol{\eta} = \mathbf{H}^{-1}\mathbf{r}. \tag{52}$$

If the matrix $\mathbf{H}$ is singular or ill-conditioned, we can replace its inverse by pseudo-inverse.

In sum, we present the multidimensional-subspace-fitting method in Algorithm 2. Interestingly, the idea of using the Taylor expansion of the projection operator is used in [25] as well.

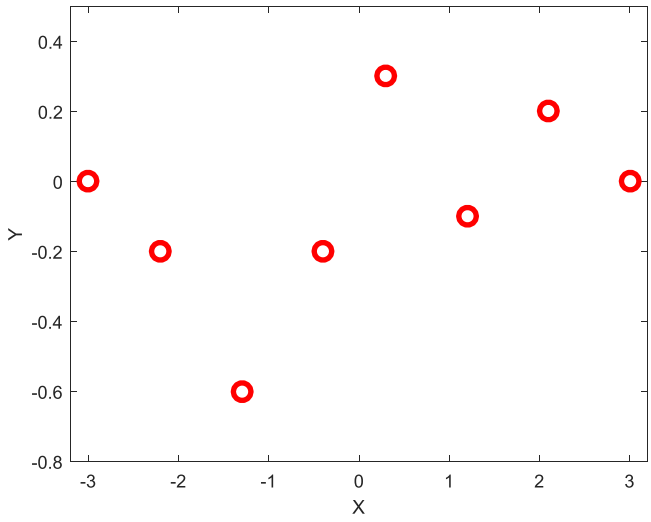
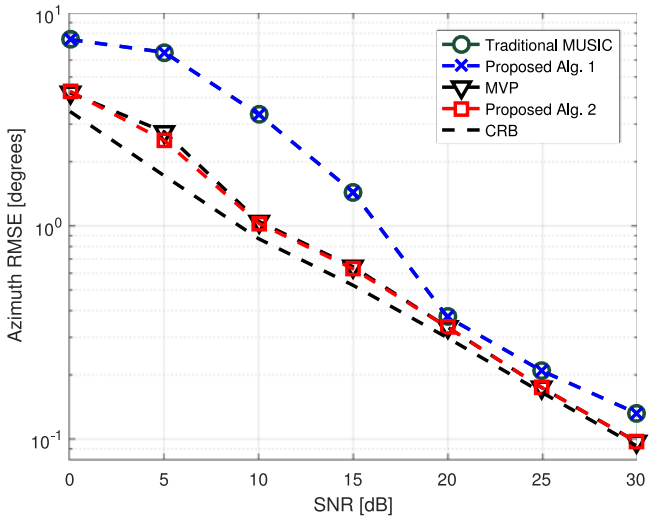
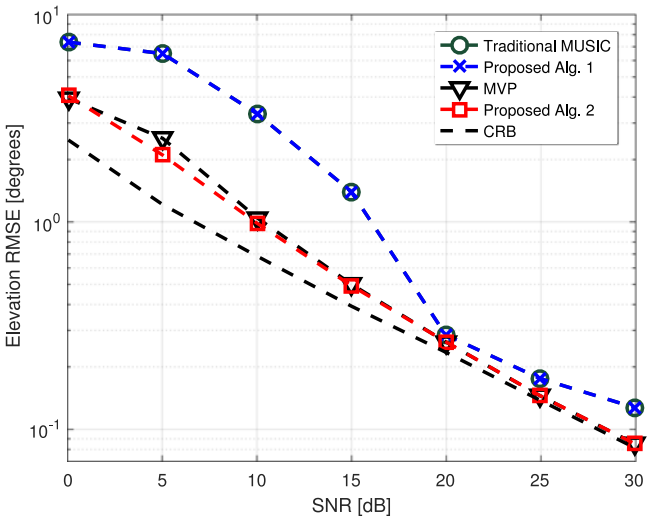


Fig. 2. Array geometry.



(a)



(b)

Fig. 3. DOA estimation RMSEs versus input SNR; uncorrelated signals; first example: a) azimuth, b) elevation.

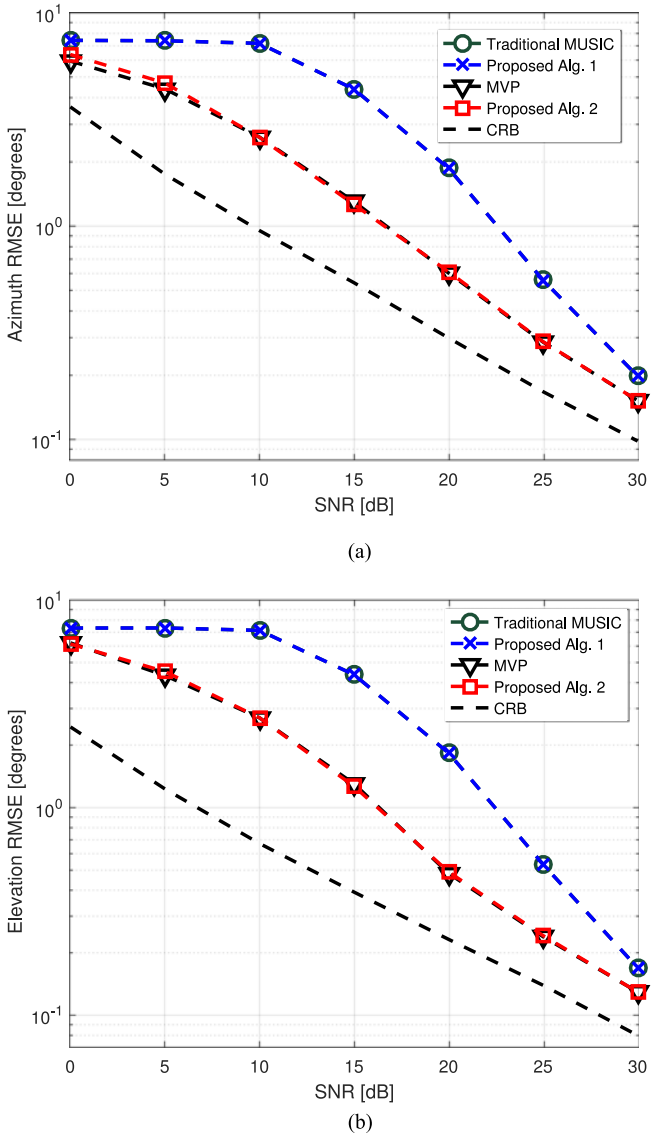


Fig. 4. DOA estimation RMSEs versus input SNR; correlated signals; first example: a) azimuth, b) elevation.

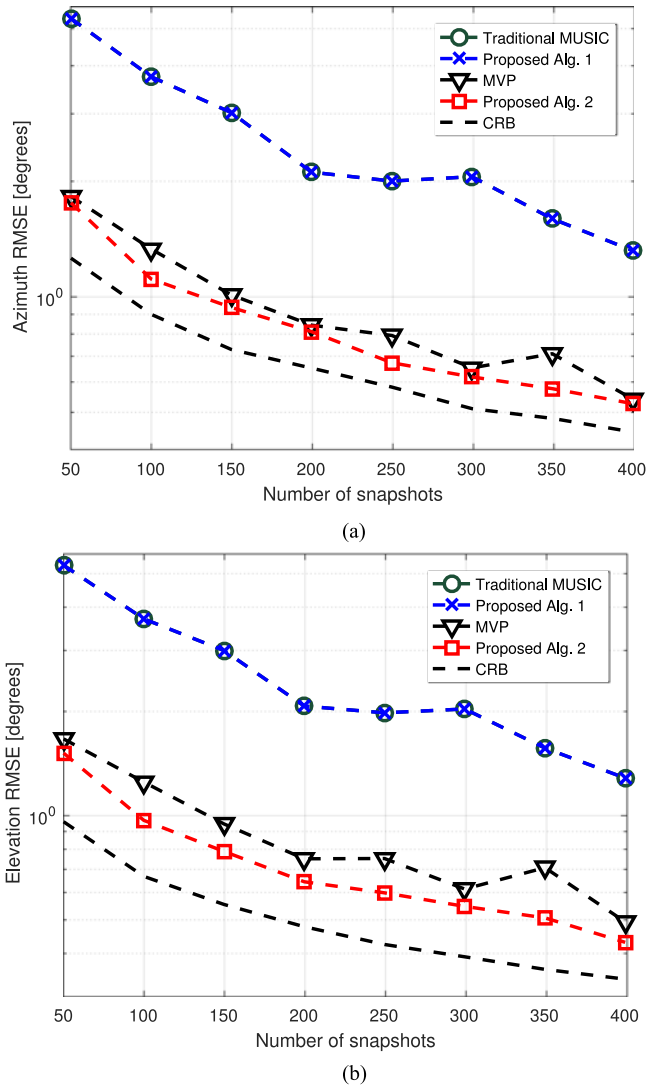


Fig. 5. DOA estimation RMSEs versus snapshot number; uncorrelated signals; second example: a) azimuth, b) elevation.

In comparison, the implementation of our method is more convenient. Additionally, it is found in the simulation part that the average of iteration times of our method is less than ten.

V. Simulation Results

In this section we evaluate the effectiveness of the proposed 2D DOA estimation approaches. Consider a nonuniform array of $L = 8$ sensors with x-y Cartesian coordinates (measured in half-wavelength) given by $\mathbf{x} = [-3.0, -2.2, -1.3, -0.4, 0.3, 1.2, 2.1, 3.0]^T$ and $\mathbf{y} = [0, -0.2, -0.6, -0.2, 0.3, -0.1, 0.2, 0]^T$, which is depicted in Fig. 2. For the sampling matrix, we choose the mode number $M = 45$ and truncated mode number $M_t = 25$. The search-point numbers are $N = N_a = 4096$ and $N_e = 2048$ which implies that the resolution is $360^\circ/4096 \approx 0.088^\circ$ for the proposed FFT-based and traditional MUSIC methods. We also assume that the array operates in the presence of $K = 2$ equally-powered signals. Throughout our simulations, 300 independent Monte Carlo runs have been used in each example. For reference, the Cramér-Rao bound (CRB) is also plotted. In the following examples, we compare our proposed algorithms with the traditional 2D MUSIC and the MVP approach presented in [11] in terms of the DOA estimation root-mean-square errors (RMSEs). Our proposed methods include the FFT-based 2D MUSIC presented in Algorithm 1 and the multidimensional-subspace-fitting method described in Algorithm 2 (in which $\mathbf{\Psi} = \widehat{\mathbf{E}}_s(\mathbf{\Lambda}_s - \sigma_n^2\mathbf{I}_K)^2\widehat{\mathbf{\Lambda}}_s^{-1}\widehat{\mathbf{E}}_s^H$ is taken). The stepsize $\mu = 0.9$ is chosen for the MVP. In addition, the MVP method uses the same initial angles as the proposed Algorithm 2.

In the first example, we compare the DOA estimation performances of the methods tested when the input signal-to-noise ratio (SNR) varies from 0 dB to 30 dB. The number of snapshots to estimate the array covariance matrix is $T = 100$. The azimuth and elevation of the first signal are randomly generated within the area $[10^\circ, 50^\circ]$ and the angles of the second signal are taken

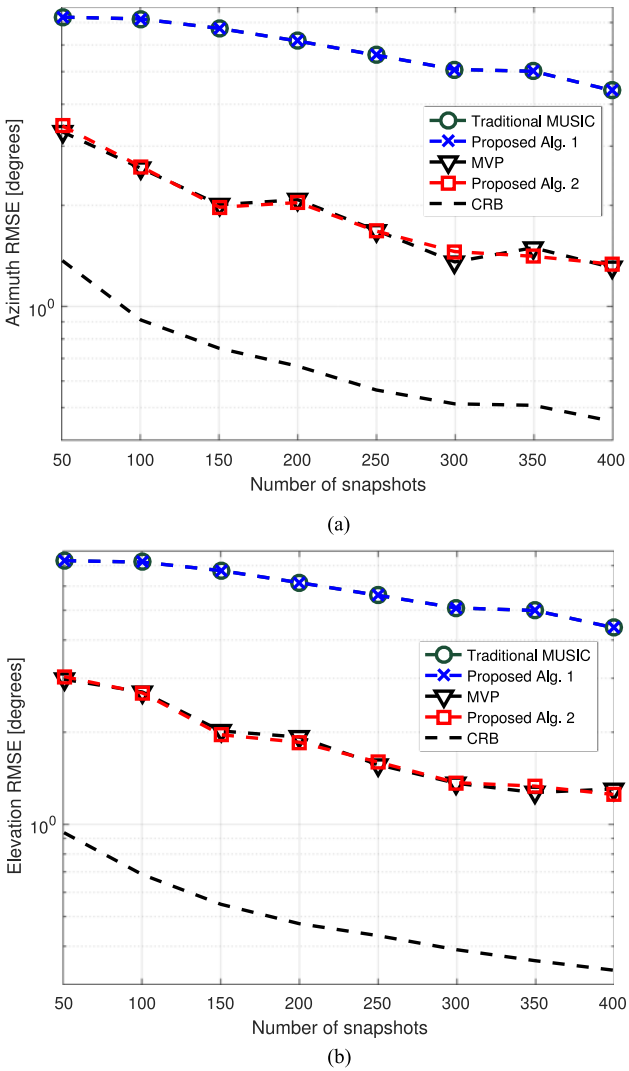


Fig. 6. DOA estimation RMSEs versus snapshot number; correlated signals; second example: a) azimuth, b) elevation.

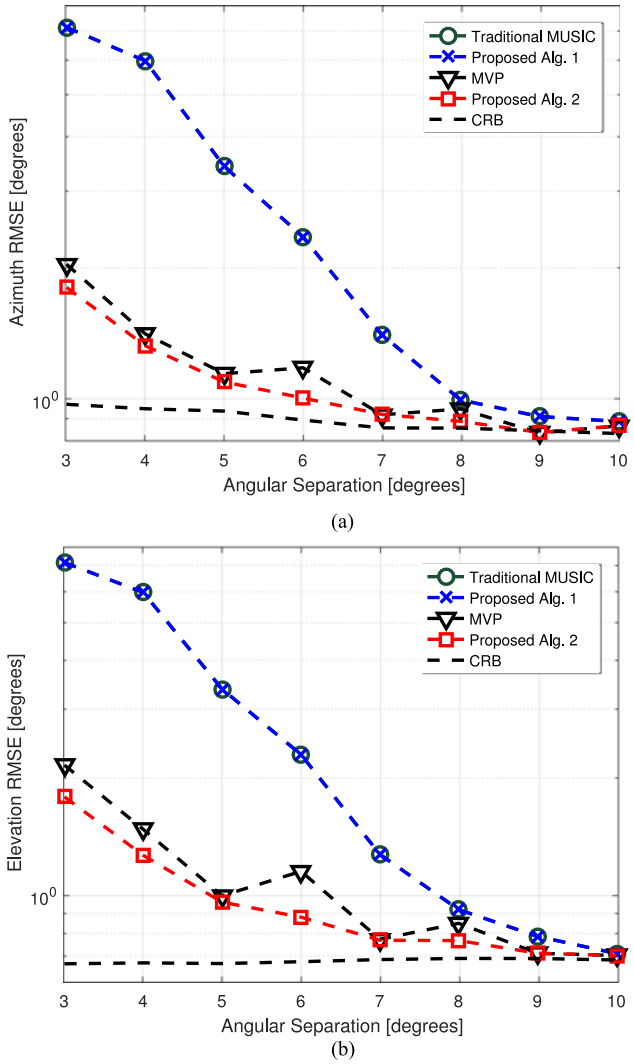


Fig. 7. DOA estimation RMSEs versus angular separation; uncorrelated signals; third example: a) azimuth, b) elevation.

as $\theta_2 = \theta_1 + 5^\circ$ and $\phi_2 = \phi_1 + 5^\circ$. This means that the angular separations between the two signals are 5° in both azimuth and elevation. The DOA estimation RMSEs are plotted in Fig. 3 for the case of uncorrelated signals, whereas Fig. 4 illustrates the case of correlated signals. Here we assume that the signal covariance matrix is $\mathbf{R}_s = 10^{(0.1\text{SNR})} \times \left[\begin{smallmatrix} 1 & 0.8 \\ 0.8 & 1 \end{smallmatrix}\right]$ when the signals are correlated. Said differently, the correlation $= 0.8$ is taken. As expected, the performance of the proposed multidimensional-subspace-fitting methods are better than the MUSIC-type methods. The performances of the traditional 2D MUSIC and the proposed FFT-based MUSIC are very similar to each other. Compared with the MVP, the proposed Algorithm 2 achieves almost the same performance with less computational complexity.

In the second example, we study the impact of the snapshot number on the methods tested. In this example, the snapshot number varies from 50 to 400, the input SNR $= 10$ dB is taken and all other parameters are chosen as before. We also depict the DOA estimation RMSEs versus the input SNR in Fig. 5 for uncorrelated signals and Fig. 6 for correlated signals. It is clear that the proposed multidimensional-subspace-fitting method outperforms MUSIC-type methods as before.

In our third example, we assume that the angular separation between the two sources changes from 3° to 10°. Here the input SNR is 10 dB and all other parameters are chosen as in the first example. Also, we examine the situations of uncorrelated and correlated signals, which are displayed in Figs. 7 and 8 respectively. As shown in these two figures, the proposed Algorithm 2 works consistently better than the traditional 2D MUSIC, with most pronounced performance improvements achieved at small source angular spacings for the uncorrelated signals.

In the last simulation experiment, we use different (randomly generated) array geometries in each simulation run. In this example, the locations of 30 array sensors are drawn uniformly from the interior of a circle the radius of which varies from 2 to 5 half-wavelengths. For the sample matrix, the mode numbers $M = [37, 41, 45, 51, 57, 63, 69]$ are chosen for the array circle radii $[2, 2.5, 3, 3.5, 4, 4.5, 5]$ half-wavelengths. The input SNR is fixed at 10 dB and all other parameters are the same as those in the first example. The truncated mode number is

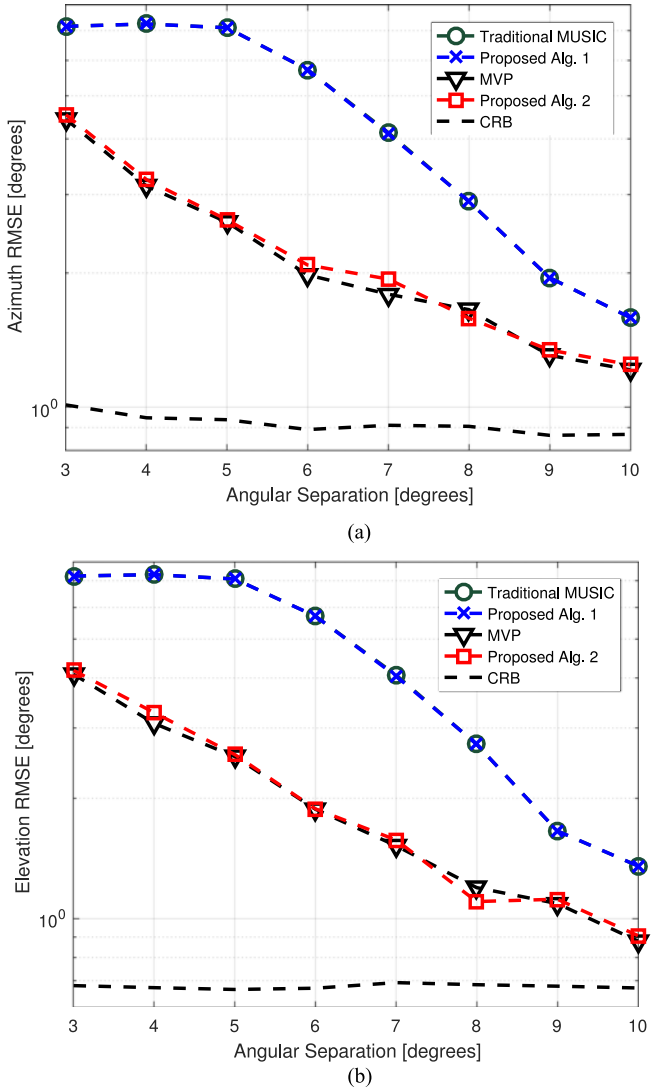


Fig. 8. DOA estimation RMSEs versus angular separation; correlated signals; third example: a) azimuth, b) elevation.

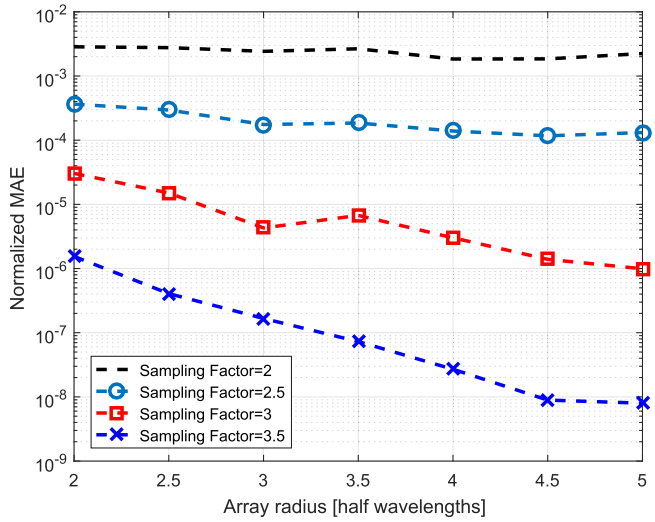


Fig. 9. Normalized mean absolute errors of 2D spatial spectrum when using different sampling factors; final example.

$M_t = \alpha\kappa R$ where the sampling factor α varies from 2 to 3.5 with step 0.5. We compare the normalized mean absolute errors (MAE) of $\frac{\|\mathbf{C}_{\text{DFT}} - \mathbf{C}_{\text{DFT}}^{(t)}\|}{\|\mathbf{C}_{\text{DFT}}\|}$ in Fig. 9 where $\mathbf{C}_{\text{DFT}}$ is the 2D spatial spectrum computed by using the full coefficient matrix whereas $\mathbf{C}_{\text{DFT}}^{(t)}$ by the truncated coefficient matrix. It can be seen from Fig. 9 that the normalized MAE of the 2D spatial spectrum can reach the level of 10^{-4} when α is no less than 2.5. As shown in the previous examples, we can achieve good performance when $M_t = 2.5\kappa R$ is chosen.

VI. Conclusion

In order to reduce the computational complexity of joint estimation of azimuth and elevation angles with arbitrary array structures, we propose an MST-based framework. First we compute a truncated coefficient matrix and then a partial 2D FFT operation is employed to calculate the 2D spatial spectrum. This algorithm is inherently parallel and the complexity is almost independent of the signal number. Then we formulate a multidimensional-subspace-fitting estimator to improve the DOA estimation. By means of the first-order Taylor expansion, we expand the projection operator around the initial angular estimates. It is found that the estimation errors correspond to the solution of a set of linear equations. Computer simulations demonstrate that the two proposed algorithms can achieve almost the same DOA estimation performance as their counterparts while ours are with less computational complexity.

Appendix A
Symmetry Properties of the Coefficients

From (12), we have the corresponding inverse DFT form

$$[\mathbf{C}]_{M+k_1,M+k_2} = \frac{1}{2M-1} \sum_{m_1=-(M-1)}^{M-1} \sum_{m_2=-(M-1)}^{M-1} \left[p(m_1\Delta\theta, m_2\Delta\phi) e^{jk_1 m_1 \Delta\theta} e^{jk_2 m_2 \Delta\phi}\right] \quad (53)$$

where $\Delta\theta = \Delta\phi = \frac{2\pi}{2M-1}$. Since the noise projector $(\mathbf{I}_L - \widehat{\mathbf{E}}_s \widehat{\mathbf{E}}_s^H)$ is nonnegative-definite, the spatial spectrum $p(\theta,\phi)$ is always a nonnegative real number. Evaluating the complex conjugate of (53) and using the fact $p(\theta,\phi) = p^*(\theta,\phi)$, we have

$$\begin{aligned}[\mathbf{C}]^*_{M+k_1,M+k_2} &= \frac{1}{2M-1} \sum_{m_1=-(M-1)}^{M-1} \sum_{m_2=-(M-1)}^{M-1} \left[p(m_1\Delta\theta, m_2\Delta\phi) e^{-jk_1 m_1 \Delta\theta} e^{-jk_2 m_2 \Delta\phi}\right] \\ &= [\mathbf{C}]_{M-k_1,M-k_2}. \end{aligned} \quad (54)$$

Moreover, when we use the 2D Effective Aperture Distribution Function (EADF) [15] method to calculate the sampling matrix $\mathbf{G}$, it is required that the manifold vector $\mathbf{a}(\theta,\phi)$ is periodic in both azimuth and elevation with period 2π. In practice, however, we measure the array responses within the area $\phi \in [0,\pi]$ only. For periodic extension, we construct the periodic calibration

$$
\begin{aligned}
[\mathbf{C}]_{M+k_1,M-k_2} &= \frac{1}{2M-1}\sum_{m_1=-(M-1)}^{M-1}\sum_{m_2=-(M-1)}^{M-1} p(m_1\Delta\theta, m_2\Delta\phi)e^{jk_1m_1\Delta\theta}e^{-jk_2m_2\Delta\phi} \\
&= \frac{1}{2M-1}\sum_{m_1=-(M-1)}^{M-1}\sum_{m_2=-(M-1)}^{M-1} p(m_1\Delta\theta+\pi, -m_2\Delta\phi)e^{-jk_1\pi}e^{jk_1(m_1\Delta\theta+\pi)}e^{-jk_2m_2\Delta\phi} \\
&= \frac{(-1)^{k_1}}{2M-1}\sum_{m_1=-(M-1)}^{M-1}\sum_{m_2=-(M-1)}^{M-1} p(m_1\Delta\theta+\pi, -m_2\Delta\phi)e^{jk_1(m_1\Delta\theta+\pi)}e^{-jk_2m_2\Delta\phi} \\
&= \frac{(-1)^{k_1}}{2M-1}\sum_{m_1=0}^{2M-1}\sum_{m_2=-(M-1)}^{M-1} p(m_1\Delta\theta, m_2\Delta\phi)e^{jk_1m_1\Delta\theta}e^{jk_2m_2\Delta\phi} \\
&= \frac{(-1)^{k_1}}{2M-1}\sum_{m_1=-(M-1)}^{M-1}\sum_{m_2=-(M-1)}^{M-1} p(m_1\Delta\theta, m_2\Delta\phi)e^{jk_1m_1\Delta\theta}e^{jk_2m_2\Delta\phi} \\
&= (-1)^{k_1}[\mathbf{C}]_{M+k_1,M+k_2} \qquad (58)
\end{aligned}
$$

matrix from the measurements by a shift of 180° in azimuth and a flip in elevation followed by discarding the first and last row to avoid the redundancy at the poles (0° and 180° elevations) [11]. This leads to

$$\mathbf{a}(\theta,\phi) = \mathbf{a}(\theta+\pi, 2\pi-\phi) = \mathbf{a}(\theta+\pi, -\phi) \qquad (55)$$

and naturally

$$p(\theta,\phi) = p(\theta+\pi, -\phi). \qquad (56)$$

Let us now turn to the coefficient $[\mathbf{C}]_{M+k_1,M-k_2}$ which can be expressed as (58), shown at the top of the page. Combining (58) and (54), we have

$$[\mathbf{C}]_{M-k_1,M+k_2} = (-1)^{k_1}\left([\mathbf{C}]_{M+k_1,M+k_2}\right)^*. \qquad (57)$$

Appendix B
Derivatives of the Projection Operator Matrix

The first derivative of the projection matrix with respect to the azimuth angle θ_k is given by [26], [28]

$$
\begin{aligned}
\frac{\partial\mathbf{\Pi}_{\boldsymbol{\eta}}}{\partial\theta_k} &= \mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\frac{\partial\mathbf{A}(\boldsymbol{\eta})}{\partial\theta_k}\mathbf{A}^{\dagger}(\boldsymbol{\eta}) + (\cdots)^H \\
&= \mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\mathbf{G}\left[0,\ldots,\frac{\partial\mathbf{v}(\theta_k)}{\partial\theta_k}\otimes\mathbf{v}(\phi_k),\ldots,0\right]\mathbf{A}^{\dagger}(\boldsymbol{\eta}) \\
&\quad +(\cdots)^H \qquad (59)
\end{aligned}
$$

where the notation $(\cdots)^H$ means that the same expression appears again with conjugate transpose and

$$\frac{\partial\mathbf{v}(\theta_k)}{\partial\theta_k} = \mathbf{v}(\theta_k)\odot\left[j\frac{M-1}{2},\ldots,1,\ldots,-j\frac{M-1}{2}\right]^T. \qquad (60)$$

Similarly, we have

$$
\begin{aligned}
\frac{\partial\mathbf{\Pi}_{\boldsymbol{\eta}}}{\partial\phi_k} &= \mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\frac{\partial\mathbf{A}(\boldsymbol{\eta})}{\partial\phi_k}\mathbf{A}^{\dagger}(\boldsymbol{\eta}) + (\cdots)^H \\
&= \mathbf{\Pi}_{\boldsymbol{\eta}}^{\perp}\mathbf{G}\left[0,\ldots,\mathbf{v}(\theta_k)\otimes\frac{\partial\mathbf{v}(\phi_k)}{\partial\phi_k},\ldots,0\right]\mathbf{A}^{\dagger}(\boldsymbol{\eta}) \\
&\quad +(\cdots)^H \qquad (61)
\end{aligned}
$$

where

$$\frac{\partial\mathbf{v}(\phi_k)}{\partial\phi_k} = \mathbf{v}(\phi_k)\odot\left[j\frac{M-1}{2},\ldots,1,\ldots,-j\frac{M-1}{2}\right]^T. \qquad (62)$$

References

[1] C. Hu, Y. Liu, H. Meng and X. Wang, "Randomized switched antenna array FMCW radar for automotive applications," *IEEE Trans. Veh. Technol.*, vol. 63, no. 8, pp. 3624–3641, Oct. 2014.

[2] R. He *et al.*, "Vehicle-to-vehicle radio channel characterization in crossroad scenarios," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 5850–5861, Aug. 2016.

[3] S. Mumtaz, J. Miquel Jornet, J. Aulin, W. H. Gerstacker, X. Dong, and B. Ai, "Terahertz communication for vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 5617–5625, Jul. 2017.

[4] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[5] M. Rübsamen and A. B. Gershman, "Direction-of-arrival estimation for nonuniform sensor arrays: From manifold separation to Fourier domain music methods," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 588–599, Feb. 2009.

[6] A. B. Gershman, M. Rübsamen, and M. Pesavento, "One- and two-dimensional direction-of-arrival estimation: An overview of search-free techniques," *Signal Process.*, vol. 90, no. 5, pp. 1338–1349, May 2010.

[7] M. A. Doron and E. Doron, "Wavefield modeling and array processing .II. Algorithms," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2560–2570, Oct. 1994.

[8] J. Zhuang, W. Li, and A. Manikas, "Fast root-music for arbitrary arrays," *Electron. Lett.*, vol. 46, no. 2, pp. 174–176, 2010.

[9] F. Belloni, A. Richter, and V. Koivunen, "DoA estimation via manifold separation for arbitrary array structures," *IEEE Trans. Signal Process.*, vol. 55, no. 10, pp. 4800–4810, Oct. 2007.

[10] J. Zhuang, W. Li, and A. Manikas, "An IDFT-based root-MUSIC for arbitrary arrays," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2010, pp. 2614–2617.

[11] M. Costa, V. Koivunen, and A. Richter, "Low complexity azimuth and elevation estimation for arbitrary array configurations," *IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2009, pp. 2185–2188.
[12] J. Zhuang, H. Xiong, W. Wang, and X. Cai, "FFT-based adaptive 2-D DOA estimation for arbitrary array structures," in *Proc. 22nd Int. Conf. Digit. Signal Process.*, Aug. 2017, pp. 1–5.
[13] J. Zhuang, J. Liu, D. Chen, and N. Yu, "Low complexity 2-D DOA estimator for arbitrary arrays: A hybrid MUSIC-based method," in *Proc. 10th Int. Conf. Inf. Commun. Signal Process.*, Dec. 2015, pp. 1–5.
[14] M. Costa, A. Richter, and V. Koivunen, "Unified array manifold decomposition based on spherical harmonics and 2-D Fourier basis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4634–4645, Sep. 2010.
[15] M. Landmann, A. Richter, and R. S. Thoma, "DoA resolution limits in MIMO channel sounding," in *Proc. Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, Jun. 2004, pp. 1708–1711.
[16] K. V. S. Babu, "A fast algorithm for adaptive estimation of root-MUSIC polynomial coefficients," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1991, pp. 2229–2232.
[17] M. A. Doron and E. Doron, "Wavefield modeling and array processing. I. Spatial sampling," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2549–2559, Oct. 1994.
[18] S. He and M. Torkelson, "Computing partial DFT for comb spectrum evaluation," *IEEE Signal Process. Lett.*, vol. 3, no. 6, pp. 173–175, Jun. 1996.
[19] H. V. Sorensen and C. S. Burrus, "Efficient computation of the DFT with only a subset of input or output points," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1184–1200, Mar. 1993.
[20] L. Deng, C. L. Yu, C. Chakrabarti, J. Kim, and V. Narayanan, "Efficient image reconstruction using partial 2D Fourier transform," in *Proc. IEEE Workshop Signal Process. Syst.*, 2008, pp. 49–54.
[21] P. A. Milder, F. Franchetti, J. C. Hoe, and M. Püschel, "Discrete Fourier transform compiler: From mathematical representation to efficient hardware," Center Silicon Syst. Implementation, Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CSSI-07–01, 2007.
[22] G. Xu and T. Kailath, "Fast subspace decomposition," *IEEE Trans. Signal Process.*, vol. 42, no. 3, pp. 539–551, Mar. 1994.
[23] F. Yan, M. Jin and X. Qiao, "Low-complexity DOA estimation based on compressed MUSIC and its performance analysis," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 1915–1930, Apr. 2013.
[24] I. Ziskind and M. Wax, "Maximum likelihood localization of multiple sources by alternating projection," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 10, pp. 1553–1560, Oct. 1988.
[25] P. Pelin, "A fast minimization technique for subspace fitting with arbitrary array manifolds," *IEEE Trans. Signal Process.*, vol. 49, no. 12, pp. 2935–2939, Dec. 2001.
[26] M. Viberg and B. Ottersten, "Sensor array processing based on subspace fitting," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1110–1121, May 1991.
[27] M. Viberg, B. Ottersten, and T. Kailath, "Detection and estimation in sensor arrays using weighted subspace fitting," *IEEE Trans. Signal Process.*, vol. 39, no. 11, pp. 2436–2449, Nov. 1991.
[28] G. H. Golub and V. Pereyra, "The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate," *SIAM J. Numer. Anal.*, vol. 10, no. 2, pp. 413–432, Apr. 1973.

Jie Zhuang is currently an Associate Professor in the School of Information and Communication Engineering at University of Electronic Science and Technology of China (UESTC). He is also with Chengdu Aerospace Communication Device Company as a Post-doctoral Fellow. He received the Ph.D. degree in electrical and electronic engineering from Imperial College London, UK, in 2011, where he was a recipient of UK/China Scholarship for Excellence Programme. He obtained his B.Eng. ('94–'98) in Chongqing University of Posts and Telecommunications, after which he worked in China Telecom (Chengdu) for nearly two years. Since 2000, he has been with UESTC, firstly as an M.Sc. student ('00–'03) and then as an academic staff member immediately after his graduation. His research interests lie in array signal processing and MIMO wireless communications.

Chenghua Duan received the B.S. degree in electronic information engineering from the Civil Aviation University of China, Tianjin, China, in 2014. Currently, he is working toward the M.S. degree at the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include direction-of-arrival estimation, source localization, and machine learning.

Wei Wang received the B.S. degree in communication engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2016. Currently, he is working toward the M.S. degree at the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include direction-of-arrival estimation, source localization, and wireless communications.

Zhi Chen received B. Eng., M. Eng., and Ph.D. degrees in electrical engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1997, 2000, and 2006, respectively. On April 2006, he joined the National Key Lab of Science and Technology on Communications (NCL), UESTC, and was a Professor with this lab from August 2013. He was a Visiting Scholar with the University of California, Riverside during 2010 and 2011. His current research interests include 5G mobile communications, tactile internet, and Terahertz communication. He has served as a Reviewer for various international journals and conferences, including the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, etc.