Near-field Source Localization in 3-D Using Two Parallel Centrally Symmetric Unfold Coprime Array

Hua Chen, *Senior Member, IEEE*, Junjie Li, Songjie Yang, Wei Liu, *Senior Member, IEEE*,
Yonina C. Eldar, *Fellow, IEEE*, and Chau Yuen, *Fellow, IEEE*

***Abstract*—Most near-field (NF) localization algorithms cannot deal with the underdetermined case, while those which can are computationally expensive due to employment of fourth-order cumulants. In this work, a low-complexity solution is provided for underdetermined three-dimensional (3-D) NF localization, by employing second-order statistics with a tailored array configuration named two parallel centrally symmetric unfold coprime (TPSC) array. Its implementation can be divided into three stages. Firstly, the proposed algorithm constructs two cross-correlation matrices based on the received array data, which eliminates the non-linear range-related information of NF signals. Secondly, covariance and vectorization operations are applied to these two cross-correlation matrices to form a virtual array with extended aperture. Finally, the two-dimensional (2-D) angle parameters are estimated by the sparse and parametric approach (SPA) and a phase retrieval operation, and then the one-dimensional (1-D) range parameter is achieved by the multiple signal classification (MUSIC) algorithm. One specific feature is that the estimated angle and range parameters are matched automatically. An analysis of the properties of the TPSC array is provided, and an optimal parameter configuration is derived, given that the total number of array elements is fixed. Simulation results demonstrate that the designed TPSC array can achieve underdetermined 3-D NF localization, and deliver enhanced estimation capabilities, surpassing those of established algorithms.**

***Index Terms*—Near-field, Coprime array, Symmetric array, Underdetermined case, Source localization.**

I. INTRODUCTION

SOURCE localization, a significant topic in array signal processing, has been extensively studied across various applications, including radar, sonar, and wireless communications [1–6]. According to the distance between the target signal and the array [7, 8], source localization can be divided into far-field (FF) source localization and near-field (NF) source localization. When the distance between the source and the array is sufficiently large, resulting in the source being situated within the FF region, the propagation delays of the FF sources are characterized by the direction-of-arrival (DOA) parameter. When the sources are situated close to the array and reside within the NF region, the impinging signals behave as spherical waves. Consequently, the NF signals are characterized by two distinct parameters: DOA and range, which renders many FF localization techniques inapplicable to NF.

With advancements in wireless communication systems, particularly the increasing array aperture, NF signal processing has garnered substantial attention due to its ability to provide more accurate channel models. Consequently, numerous new techniques related to NF localization have been developed, such as NF beam training [9, 10] and NF channel estimation [11–13]. These advancements underscore the significance of NF signal processing. One approach to increase the array aperture is the utilization of sparse linear arrays (SLAs) for NF source localization [14–21]. Compared to uniform linear arrays (ULAs), SLAs [22–27] offer a larger physical aperture and more degrees of freedom (DOFs) for a given number of elements, and therefore play an essential role for underdetermined direction of arrival estimation.

For SLA-based NF source localization in [14], an algorithm based on a sparse symmetric array is presented, leveraging a hybrid-order multiple signal classification (MUSIC) algorithm to process the cumulant matrix obtained from the array, and thereby estimating the required angle and range parameters. With partial knowledge of calibration sources, an iterative self-calibration algorithm is introduced in [15], facilitating concurrent estimation of gain and phase error parameters, as well as source parameters. In [16], a parameter estimation method for NF sources using symmetric nested array (SNA) is suggested, where the application of fourth-order cumulants in the received data allows the SNA to form at least $4N + 4MN - 2M - 3$ continuous segments with only $2M + 2N - 1$ sensors (M and N are integers, with $N > 1$ and $M > 0$). Based on a symmetric linear array in [17], oblique projection techniques

This work was supported by the Zhejiang Provincial Natural Science Foundation of China under Grants LY23F010003 and LR20F010001, by the “Pioneer” and “Leading Goose” R & D Program of Zhejiang Province under Grant 2024C01105, by the National Natural Science Foundation of China under Grants U24A20217 and 62001256, and the UK Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/V009419/2, and by the Ministry of Education Singapore MOE Tier 2 under Grant MOE-T2EP50220-0019, and by National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research & Development Programme under Grant FCP-NTU-RG-2024-025, and by the China Scholarship Council under Grants 202408330215 and 202306070094. For the purpose of open access, the author(s) has applied a Creative Commons Attribution (CC BY) license to any Accepted Manuscript version arising. (*Corresponding author: Hua Chen.*)

Hua Chen and Junjie Li are with the Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China. (e-mail: dkchenhua0714@hotmail.com)

Songjie Yang is with the National Key Laboratory of Wireless Communications, University of Electronic Science and Technology of China, Chengdu 610054, China. (e-mail: yangsongjie@std.uestc.edu.cn)

Wei Liu is with the Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong. (e-mail: wei2.liu@polyu.edu.hk)

Yonina C. Eldar is with the Faculty of Mathematics and Computer Science, Weizmann Institute of Science, Rehovot 7610001, Israel. (e-mail: yonina.eldar@weizmann.ac.il)

Chau Yuen is with the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore 639798, Singapore. (e-mail: chau.yuen@ntu.edu.sg)

are applied to separate NF sources from mixed near and FF sources. Atomic norm minimization is then used for angle parameter estimation, followed by range parameter estimation using a one-dimensional (1-D) search method. In [18], a convolutional neural network (CNN) is proposed to transform the received data from time domain to frequency domain, employing CNN and autoencoder to separate and locate the mixed sources. A symmetric displaced coprime array (SDCA) is employed in [19], composed of three SLAs, with the same number of array elements, which can provide a larger array aperture and a wider range of continuous segments than SNA in [16]. In [20], angle and range parameters are estimated through spatial smoothing and the MUSIC algorithm, utilizing a symmetric extended nested array (SENA) built upon the foundation developed from the compressed symmetric nested array (CSNA). With the optimization of array configuration parameters, SENA exhibits larger physical aperture, longer continuous segments, and better estimation performance than existing symmetric sparse arrays. Also based on SLA, a NF source localization algorithm is introduced in [21], leveraging the structure of a coprime array. The method first constructs an off-grid model which solely contains angle parameter, and subsequently, an iterative approach is employed for estimating the 1-D angle parameter, which are then substituted into the range parameter model for iteratively estimating the range parameter.

Although the above algorithms based on SLA yield excellent performance in parameter estimation, they are limited to two-dimensional (2-D) parameter estimation for the NF signals case, namely, the 1-D azimuth and range. In practice, it is often desirable to locate NF targets in three-dimensional (3-D) space with planar arrays, which involves estimating azimuth, elevation, and range parameters, thereby posing a more intricate challenge in parameter pairing.

To enable 3-D parameter estimation for NF signals, various arrays have been proposed, such as cross arrays, uniform circular arrays, and rectangular arrays. As a representative example, a gridless NF source localization algorithm is proposed based on a cross array in [28], which constructs a sparse model to transform the parameter estimation problem into a semi-positive definite programming (SDP) problem. In [29], a uniform circular array is suggested utilizing a joint phase interferometer and the MUSIC algorithm. A NF source localization algorithm for uniform rectangular array (URA) or centrally symmetric sparse rectangular array (SRA) is developed in [30] which utilizes fourth-order cumulants to convert the angle estimation problem into a low-rank reconstruction problem, and achieve range estimation using a 1-D MUSIC algorithm. Similar to [28], cross array is also used in [31] for NF source localization, yet without resorting to the Fresnel approximation. Accounting for disparate propagation attenuation across sensors, the algorithm constructs a third-order parallel factor (PARAFAC) data model, and then the parameters are estimated through trilinear decomposition and a least squares method.

In recent study of nonuniform cross arrays, fourth-order cumulants matrices and vectorization operations are employed in [32], effectively achieving underdetermined estimation for 2-D angle and range parameters of NF sources. Although the aforementioned algorithms are capable of achieving NF source localization, they cannot work in the underdetermined case (except for Ref. [32]). Research into underdetermined estimation algorithms for array processing is instrumental in reducing the cost associated with a large number of array elements.

In summary, existing NF algorithms can be categorized into three types: the first type ([14]-[21]) offers high accuracy in parameter estimation but is limited to the 2-D plane scene; the second type ([28]-[31]) achieves 3-D parameter estimation but lacks the capability for underdetermined estimation; the third type ([32]), though capable of underdetermined estimation, relies on fourth-order cumulants, resulting in high computational complexity. These challenges underscore the pressing need for low-complexity solutions for underdetermined 3-D NF estimation.

In this paper, we propose an algorithm based on a two parallel centrally symmetric unfold coprime (TPSC) array for underdetermined localization of NF sources. Based on the special array structure, the proposed algorithm eliminates the non-linear range-related information in the phase part of the steering matrix by constructing two cross-correlation matrices. Subsequently, it enlarges the array aperture through covariance and vectorization operations. Finally, the NF signals' 3-D parameters are estimated using the sparse and parametric approach (SPA) [33] and the 1-D MUSIC algorithm.

The main contributions of this paper are as follows:

1) A TPSC array is proposed, which unfolds the coprime array and introduces central symmetry, facilitating elimination of the nonlinear range-related information inherent in the phase component of the steering matrix for NF signals. An analysis of the TPSC array's properties is provided, followed by proposition of the optimal parameter configuration for the proposed TPSC array.

2) The cross-correlation calculation is utilized to construct two matrices followed by vectorization, which expands the virtual array aperture, enhancing the parameter estimation performance. Simulation results show the proposed algorithm's superiority in terms of estimation accuracy, underdetermined and mixed source estimation abilities.

Notations: $(\cdot)^*$, $(\cdot)^{\mathrm{T}}$, $(\cdot)^\dagger$, and $(\cdot)^{\mathrm{H}}$ respectively denote the operations of conjugation, transpose, pseudo-inverse, and conjugate transpose; $\mathrm{E}\{\cdot\}$ represents the mathematical expectation; $\odot$ denotes the Khatri-Rao product; $\mathrm{vec}(\cdot)$ represents vectorization; $\mathrm{diag}(\cdot)$ stands for diagonalization; $\mathrm{undiag}(\cdot)$ represents diagonal extraction; $\mathrm{angle}(\cdot)$ gives the the phase/angle of a complex number, and $\mathrm{tr}(\cdot)$ represents the trace of a matrix.

II. PROPOSED TPSC ARRAY

Consider an NF scenario in which K narrowband NF sources are uniquely described by the 3-D parameters: azimuth φ_k, elevation θ_k, and range r_k, as shown in Fig. 1, where $k = 1, 2, \cdots, K$. The xoy plane of the coordinate system is populated with sensors, where the sensor at origin is set as the reference. The unit length along all axes within the coordinate system is defined as d, where $d = \lambda/4$ and λ represents the signal wavelength. Specifically, for a sensor located at (m,n) in the xoy plane, the received signal is modeled as follows

$$x_{m,n}(t)=\sum_{k=1}^{K}s_k(t)e^{j\tau_{m,n}(k)}+v_{m,n}(t) \tag{1}$$

where $s_k(t)$ is the NF signals and $v_{m,n}(t)$ denotes the additive white Gaussian noise. The k-th signal's path delay with respect to the sensor located at (m,n) is $\tau_{m,n}(k)$, which can be expressed as

$$\tau_{m,n}(k)=\frac{2\pi}{\lambda}[r_{m,n}(k)-r_k] \tag{2}$$

where $r_{m,n}(k)$ is the range from the sensor located at (m,n) to the k-th signal and r_k is the range from coordinate origin to the k-th signal. By geometric principles,

$$r_{m,n}(k)=\sqrt{(r_k sin\theta_k cos\varphi_k-md)^2+(r_k sin\theta_k sin\varphi_k-nd)^2+(r_k sin\theta_k)^2} \tag{3}$$

Applying the Fresnel approximation [34–36] to $r_{m,n}(k)$ and substituting it into (2), we obtain the following expression

$$\tau_{m,n}(k)\approx\omega_{kx}m+\phi_{kx}m^2+\omega_{ky}n+\phi_{ky}n^2+\mu_k \tag{4}$$

where

$$\omega_{kx}=\frac{-2\pi d sin\theta_k cos\varphi_k}{\lambda}=\frac{-2\pi d cos\alpha_k}{\lambda} \tag{5}$$

$$\phi_{kx}=\frac{\pi d^2(1-sin^2\theta_k cos^2\varphi_k)}{\lambda r_k}=\frac{\pi d^2 sin^2\alpha_k}{\lambda r_k} \tag{6}$$

$$\omega_{ky}=\frac{-2\pi d sin\theta_k sin\varphi_k}{\lambda}=\frac{-2\pi d cos\beta_k}{\lambda} \tag{7}$$

$$\phi_{ky}=\frac{\pi d^2(1-sin^2\theta_k sin^2\varphi_k)}{\lambda r_k}=\frac{\pi d^2 sin^2\beta_k}{\lambda r_k} \tag{8}$$

$$\mu_k=\frac{-\pi d^2 sin^2\theta_k sin2\varphi_k}{\lambda r_k}=\frac{-2\pi d^2 cos\alpha_k cos\beta_k}{\lambda r_k}. \tag{9}$$

In (5)-(9), we transform the challenge of solving for angles θ_k and φ_k into angles α_k and β_k. Consequently, (1) can be reformulated as

$$x_{m,n}(t)=\sum_{k=1}^{K}s_k(t)e^{j[\omega_{kx}m+\phi_{kx}m^2+\omega_{ky}n+\phi_{ky}n^2+\mu_k]}+v_{m,n}(t). \tag{10}$$

As indicated by (10), the 3-D parameter estimation problem is fundamentally nonlinear, arising from the exponential term involving range-related nonlinear information such as ϕ_{kx} and ϕ_{ky}. To counteract this challenge, we design the TPSC array and apply cross-correlation calculations to the received data, effectively eliminating the undesirable nonlinear components.

The TPSC model consists of two subarrays and one sensor set at origin, where subarray 1 and subarray 2 have identical configurations and exhibit centrosymmetry about the origin. Each subarray consists of two ULAs, denoted as ULA 1 and ULA 2. The number of elements in ULA 1 and ULA 2 are M_1 and M_2, respectively, where M_1 and M_2 are coprime integers with $M_1>M_2$. ULA 1 and ULA 2 share elements on the x-axis. The received data can be respectively represented as

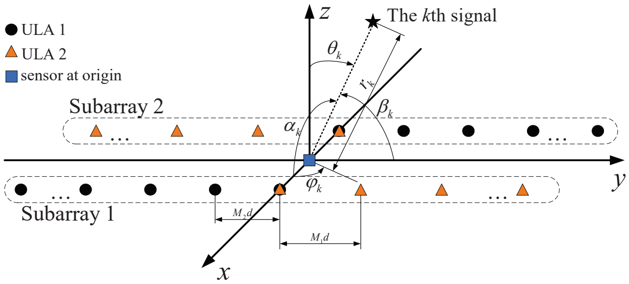


Fig. 1. The two parallel centrally symmetric unfold coprime array structure.

$$\mathbf{x}_1(t)=\mathbf{A}_1(\alpha,\beta,r)\mathbf{s}(t)+\mathbf{n}_1(t) \qquad t\in[L] \tag{11}$$

$$\mathbf{x}_2(t)=\mathbf{A}_2(\alpha,\beta,r)\mathbf{s}(t)+\mathbf{n}_2(t) \qquad t\in[L] \tag{12}$$

$$x_3(t)=\sum_{k=1}^{K}s_k(t)+n_3(t) \qquad t\in[L] \tag{13}$$

where t is the snapshot index and L is the number of snapshots. Here, $\mathbf{x}_1(t)=[x_{1,1}(t),x_{1,2}(t),\cdots,x_{1,(M_1+M_2-1)}(t)]^{\mathrm{T}}$, $\mathbf{x}_2(t)=[x_{2,1}(t),x_{2,2}(t),\cdots,x_{2,(M_1+M_2-1)}(t)]^{\mathrm{T}}$, and $x_3(t)$ is the received data of the element at the origin, with $x_{1,a}(t)$ and $x_{2,b}(t)$ denoting the received data of the ath element in subarray 1 and the bth element in subarray 2 ($a,b\in[1,M_1+M_2-1]$), respectively. $\mathbf{A}_1(\alpha,\beta,r)=[\mathbf{a}_1(\alpha_1,\beta_1,r_1),\mathbf{a}_1(\alpha_2,\beta_2,r_2),\cdots,\mathbf{a}_1(\alpha_K,\beta_K,r_K)]\in\mathbb{C}^{(M_1+M_2-1)\times K}$, $\mathbf{A}_2(\alpha,\beta,r)=[\mathbf{a}_2(\alpha_1,\beta_1,r_1),\mathbf{a}_2(\alpha_2,\beta_2,r_2),\cdots,\mathbf{a}_2(\alpha_K,\beta_K,r_K)]\in\mathbb{C}^{(M_1+M_2-1)\times K}$ are the steering matrices of the subarrays with each element in the corresponding column being $a_{1m}(\alpha_k,\beta_k,r_k)=e^{-j\tau_{1m,k}}$, $a_{2n}(\alpha_k,\beta_k,r_k)=e^{-j\tau_{2n,k}}$, where $1\leqslant m,n\leqslant(M_1+M_2-1)$, and $\tau_{1m,k}$ and $\tau_{2n,k}$ respectively represent the propagation delays associated with the mth element in subarray 1 and nth element in subarray 2. $\mathbf{s}(t)=[s_1(t),s_2(t),\cdots,s_K(t)]^{\mathrm{T}}$ is the signal vector, and $\mathbf{n}_1(t)$, $\mathbf{n}_2(t)$, $n_3(t)$ denote the additive white Gaussian noise of the corresponding subarrays. For convenience, the steering matrices of subarrays 1 and 2 are simplified into $\mathbf{A}_1$ and $\mathbf{A}_2$.

Define $\mathbf{\Omega}_1=[\Omega_{1,1},\Omega_{1,2},\cdots,\Omega_{1,(M_1+M_2-1)}]$ and $\mathbf{\Omega}_2=[\Omega_{2,1},\Omega_{2,2},\cdots,\Omega_{2,(M_1+M_2-1)}]$ as the sets of y-coordinate values of all elements in subarray 1 and subarray 2, respectively, where $(0.5,\Omega_{1,a},0)$ and $(-0.5,\Omega_{2,b},0)$ denote the coordinate of the ath element in subarray 1 and the bth element in subarray 2 ($a,b\in[1,M_1+M_2-1]$). Due to the centrosymmetric property of subarray 1 and subarray 2 with respect to the origin, we have $\Omega_{1,a}=-\Omega_{2,b}$, where $a+b=M_1+M_2$.

Considering the kth column of $\mathbf{A}_1$ and $\mathbf{A}_2$, we have

$$\begin{aligned}\mathbf{a}_1(\alpha_k,\beta_k,r_k) &= \\ &= \big[e^{-j[0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{1,1}\omega_{ky}+\Omega_{1,1}^2\phi_{ky}+0.5\Omega_{1,1}\mu_k]}, \\ &\quad e^{-j[0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{1,2}\omega_{ky}+\Omega_{1,2}^2\phi_{ky}+0.5\Omega_{1,2}\mu_k]}, \\ &\quad \cdots, e^{-j[0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{1,P}\omega_{ky}+\Omega_{1,P}^2\phi_{ky}+0.5\Omega_{1,P}\mu_k]}\big],\end{aligned} \tag{14}$$

and

$$\begin{aligned}\mathbf{a}_2(\alpha_k,\beta_k,r_k) &= \\ &= \big[e^{-j[-0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{2,1}\omega_{ky}+\Omega_{2,1}^2\phi_{ky}-0.5\Omega_{2,1}\mu_k]}, \\ &\quad e^{-j[-0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{2,2}\omega_{ky}+\Omega_{2,2}^2\phi_{ky}-0.5\Omega_{2,2}\mu_k]}, \\ &\quad \cdots, e^{-j[-0.5\omega_{kx}+0.5^2\phi_{kx}+\Omega_{2,P}\omega_{ky}+\Omega_{2,P}^2\phi_{ky}-0.5\Omega_{2,P}\mu_k]}\big],\end{aligned} \tag{15}$$

where the parameter P in $\Omega_{1,P}$ and $\Omega_{2,P}$ is equal to $M_1 + M_2 - 1$, $\omega_{kx} = -2\pi d\cos(\alpha_k)/\lambda$, $\phi_{kx} = \pi d^2\sin^2(\alpha_k)/\lambda r_k$, $\omega_{ky} = -2\pi d\cos(\beta_k)/\lambda$, $\phi_{ky} = \pi d^2\sin^2(\beta_k)/\lambda r_k$, and $\mu_k = -2\pi d^2\cos(\alpha_k)\cos(\beta_k)/\lambda r_k$.

As the range parameter is taken to $+\infty$, the steering vector of the NF model in (14) and (15) will degenerate to that of the FF one, in which the proposed method can still work, as shown in Sec. V-F.

III. THE PROPOSED ALGORITHM

This section presents an algorithmic flow where, starting from the data in (11), (12) and (13), cross-correlation matrices in (30) and (31) are constructed, which serve to form virtual FF data in (36) and (37), respectively. Then, SPA is used for sparse recovery in (39) to determine the angle β_k, followed by phase retrieval to obtain the angle α_k. Finally, the MUSIC algorithm is applied to (49) to estimate the range r_k. The proposed algorithm is illustrated in the flowchart shown in Fig. 2, providing a clear visualization of its procedural steps.

A. Construction of Cross-Correlation Matrices

In this part, we construct two cross-correlation matrices using the data received by subarray 1 and subarray 2. The expressions for two collections of spatial-domain correlations at lag τ are given as

$$\begin{aligned}r_{x_{2,l},x_{1,G-l}}(\tau) &= \mathrm{E}\big\{x_{2,l}(t+\tau)x_{1,G-l}^*(t)\big\} \\ &= \sum_{k=1}^{K} r_{sk}(\tau)e^{j(\omega_{kx}-2\Omega_{2,l}\omega_{ky})}\end{aligned} \tag{16}$$

$$\begin{aligned}r_{x_{1,l},x_{2,G-l}}(\tau) &= \mathrm{E}\big\{x_{1,l}(t+\tau)x_{2,G-l}^*(t)\big\} \\ &= \sum_{k=1}^{K} r_{sk}(\tau)e^{j(-\omega_{kx}-2\Omega_{1,l}\omega_{ky})}\end{aligned} \tag{17}$$

where $G = M_1 + M_2$, and $r_{sk}(\tau) = E\big\{s_k(t+\tau)s_k^*(t)\big\}$, $l = 1, 2, \cdots, (M_1 + M_2 - 1)$, $\tau \in N$, with N representing the number of pseudo snapshots. Arranging $r_{x_{2,l},x_{1,G}}(\tau)$ and $r_{x_{1,l},x_{2,G}}(\tau)$ according to l allows the creation of two column vectors, denoted as $\mathbf{r}_1(\tau)$ and $\mathbf{r}_2(\tau)$:

$$\mathbf{r}_1(\tau) = [r_{x_{2,1},x_{1,G-1}}(\tau), r_{x_{2,2},x_{1,G-2}}(\tau), \cdots, r_{x_{2,G-1},x_{1,1}}(\tau)]^{\mathrm{T}} \tag{18}$$

$$\mathbf{r}_2(\tau) = [r_{x_{1,1},x_{2,G-1}}(\tau), r_{x_{1,2},x_{2,G-2}}(\tau), \cdots, r_{x_{1,G-1},x_{2,1}}(\tau)]^{\mathrm{T}}. \tag{19}$$

Next, $\mathbf{r}_1(\tau)$ and $\mathbf{r}_2(\tau)$ can be written as

$$\mathbf{r}_1(\tau) = \mathbf{B}_1\mathbf{\Phi}\mathbf{r}_s(\tau) \tag{20}$$

$$\mathbf{r}_2(\tau) = \mathbf{B}_2\mathbf{\Phi}^*\mathbf{r}_s(\tau) \tag{21}$$

where

$$\mathbf{B}_1 = \big[\mathbf{b}_1(\omega_{1y}), \mathbf{b}_1(\omega_{2y}), \cdots, \mathbf{b}_1(\omega_{Ky})\big] \tag{22}$$

$$\mathbf{B}_2 = \big[\mathbf{b}_2(\omega_{1y}), \mathbf{b}_2(\omega_{2y}), \cdots, \mathbf{b}_2(\omega_{Ky})\big] \tag{23}$$

$$\begin{aligned}\mathbf{b}_1(\omega_{ky}) &= \big[e^{-2j\Omega_{2,1}\omega_{ky}}, e^{-2j\Omega_{2,2}\omega_{ky}}, \\ &\quad \cdots, e^{-2j\Omega_{2,(M_1+M_2-1)}\omega_{ky}}\big]^{\mathrm{T}}\end{aligned} \tag{24}$$

$$\begin{aligned}\mathbf{b}_2(\omega_{ky}) &= \big[e^{-2j\Omega_{1,1}\omega_{ky}}, e^{-2j\Omega_{1,2}\omega_{ky}}, \\ &\quad \cdots, e^{-2j\Omega_{1,(M_1+M_2-1)}\omega_{ky}}\big]^{\mathrm{T}}\end{aligned} \tag{25}$$

$$\mathbf{\Phi} = \mathrm{diag}([e^{j\omega_{1x}}, e^{j\omega_{2x}}, \cdots, e^{j\omega_{Kx}}]) \tag{26}$$

$$\mathbf{r}_s(\tau) = [r_{s1}(\tau), r_{s2}(\tau), \cdots, r_{sK}(\tau)]^{\mathrm{T}} \tag{27}$$

where $\mathbf{B}_1$ and $\mathbf{B}_2$ are both $(M_1 + M_2 - 1)\times K$ matrices. By collecting N pseudo snapshots, we have

$$\mathbf{R}_1 = [\mathbf{r}_1(1), \mathbf{r}_1(2), \cdots, \mathbf{r}_1(N)] \tag{28}$$

$$\mathbf{R}_2 = [\mathbf{r}_2(1), \mathbf{r}_2(2), \cdots, \mathbf{r}_2(N)], \tag{29}$$

with

$$\mathbf{R}_1 = \mathbf{B}_1\mathbf{\Phi}\mathbf{R}_s \tag{30}$$

$$\mathbf{R}_2 = \mathbf{B}_2\mathbf{\Phi}^*\mathbf{R}_s \tag{31}$$

and $\mathbf{R}_1$ and $\mathbf{R}_2$ are $(M_1 + M_2 - 1)\times N$ matrices, and $\mathbf{R}_s = [\mathbf{r}_s(1), \mathbf{r}_s(2), \cdots, \mathbf{r}_s(N)]$.

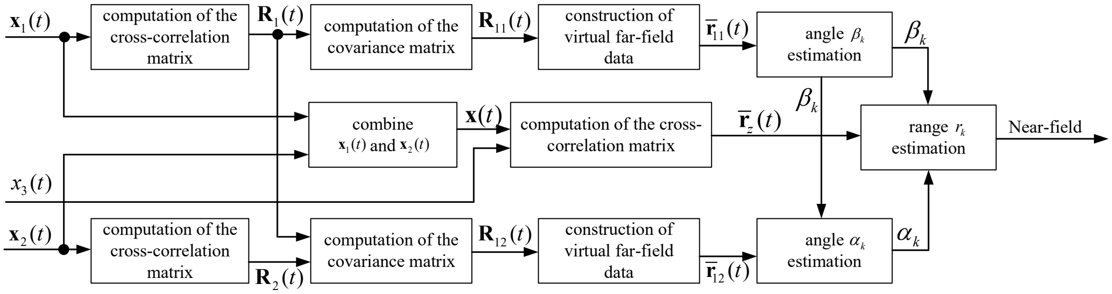


Fig. 2. The simplified flowchart of the proposed algorithm.

B. Construction of Virtual Far-field Data

This section focuses on constructing virtual far-field data in (36), (37) and (38), which enable forming a sparse recovery problem and a phase retrieval formulation for the subsequent estimates of the angle β_k and α_k in part C of Sec. III, respectively.

First, we construct two covariance matrices as follows

$$\begin{aligned}\mathbf{R}_{11} &= \mathrm{E}\{\mathbf{R}_1\mathbf{R}_1^{\mathrm{H}}\} \\ &= \mathrm{E}\{\mathbf{B}_1\mathbf{\Phi}\mathbf{R}_s\mathbf{R}_s^{\mathrm{H}}\mathbf{\Phi}^*\mathbf{B}_1^{\mathrm{H}}\} \\ &= \mathbf{B}_1\mathbf{R}_{ss}\mathbf{B}_1^{\mathrm{H}}\end{aligned} \tag{32}$$

$$\begin{aligned}\mathbf{R}_{12} &= \mathrm{E}\{\mathbf{R}_1\mathbf{R}_2^{\mathrm{H}}\} \\ &= \mathrm{E}\{\mathbf{B}_1\mathbf{\Phi}\mathbf{R}_s\mathbf{R}_s^{\mathrm{H}}\mathbf{\Phi}\mathbf{B}_2^{\mathrm{H}}\} \\ &= \mathbf{B}_1\mathbf{\Phi}^2\mathbf{R}_{ss}\mathbf{B}_2^{\mathrm{H}}\end{aligned} \tag{33}$$

where $\mathbf{\Phi}^2$ is the square of matrix $\mathbf{\Phi}$, and $\mathbf{R}_{ss} = \mathrm{E}\{\mathbf{R}_s\mathbf{R}_s^{\mathrm{H}}\} = \mathrm{diag}([\sigma_1^2, \sigma_2^2, \cdots, \sigma_K^2])$ is a diagonal matrix. The kth diagonal element of $\mathbf{R}_{ss}$ is the power of the kth incident source.

Second, by vectorizing $\mathbf{R}_{11}$ and $\mathbf{R}_{12}$, we have

$$\begin{aligned}\tilde{\mathbf{r}}_{11} &= \mathrm{vec}(\mathbf{R}_{11}) \\ &= (\mathbf{B}_1^* \odot \mathbf{B}_1)\mathrm{undiag}(\mathbf{R}_{ss}) \\ &= \tilde{\mathbf{B}}_{11}\mathrm{undiag}(\mathbf{R}_{ss})\end{aligned} \tag{34}$$

$$\begin{aligned}\tilde{\mathbf{r}}_{12} &= \mathrm{vec}(\mathbf{R}_{12}) \\ &= (\mathbf{B}_2^* \odot \mathbf{B}_1)\mathrm{undiag}(\mathbf{\Phi}^2) \\ &= \tilde{\mathbf{B}}_{21}\mathrm{undiag}(\mathbf{\Phi}^2\mathbf{R}_{ss})\end{aligned} \tag{35}$$

where $\tilde{\mathbf{B}}_{11} = [\tilde{\mathbf{b}}_{11}(\omega_{1y}), \tilde{\mathbf{b}}_{11}(\omega_{2y}), \cdots, \tilde{\mathbf{b}}_{11}(\omega_{Ky})]$ with $\tilde{\mathbf{b}}_{11}(\omega_{ky}) = \mathbf{b}_1^*(\omega_{ky}) \odot \mathbf{b}_1(\omega_{ky})$, and $\tilde{\mathbf{B}}_{21} = [\tilde{\mathbf{b}}_{21}(\omega_{1y}), \tilde{\mathbf{b}}_{21}(\omega_{2y}), \cdots, \tilde{\mathbf{b}}_{21}(\omega_{Ky})]$ with $\tilde{\mathbf{b}}_{21}(\omega_{ky}) = \mathbf{b}_2^*(\omega_{ky}) \odot \mathbf{b}_1(\omega_{ky})$. However, the column vectors $\tilde{\mathbf{r}}_{11}$ and $\tilde{\mathbf{r}}_{12}$ contain numerous redundant elements, necessitating a process of redundancy removal and reordering, after which we have

$$\bar{\mathbf{r}}_{11} = \overline{\mathbf{B}}_{11}\mathrm{undiag}(\mathbf{R}_{ss}) \tag{36}$$

$$\bar{\mathbf{r}}_{12} = \overline{\mathbf{B}}_{21}\mathrm{undiag}(\mathbf{\Phi}^2\mathbf{R}_{ss}) \tag{37}$$

where $\overline{\mathbf{B}}_{11} = [\overline{\mathbf{b}}_{11}(\omega_{1y}), \overline{\mathbf{b}}_{11}(\omega_{2y}), \cdots, \overline{\mathbf{b}}_{11}(\omega_{Ky})]$ with $\overline{\mathbf{b}}_{11}(\omega_{ky}) = [e^{-2j\Omega_{3,1}\omega_{ky}}, e^{-2j\Omega_{3,2}\omega_{ky}}, \cdots, e^{-2j\Omega_{3,n_1}\omega_{ky}}]^{\mathrm{T}}$, and $\overline{\mathbf{B}}_{21} = [\overline{\mathbf{b}}_{21}(\omega_{1y}), \overline{\mathbf{b}}_{21}(\omega_{2y}), \cdots, \overline{\mathbf{b}}_{21}(\omega_{Ky})]$ with $\overline{\mathbf{b}}_{21}(\omega_{ky}) = [e^{-2j\Omega_{4,1}\omega_{ky}}, e^{-2j\Omega_{4,2}\omega_{ky}}, \cdots, e^{-2j\Omega_{4,n_2}\omega_{ky}}]^{\mathrm{T}}$; Ω_{3,n_1} and Ω_{4,n_2} represent the y-coordinate value of the n_1th element in the virtual array 1 and the n_2th element in the virtual array 2.

The covariance matrix of $\bar{\mathbf{r}}_{11}$ can be calculated as follows

$$\mathbf{R} = \mathrm{E}\{\bar{\mathbf{r}}_{11}\bar{\mathbf{r}}_{11}^{\mathrm{H}}\} = \mathbf{\Gamma}\mathbf{T}\mathbf{\Gamma}^{\mathrm{H}}. \tag{38}$$

Here $\mathbf{T} = \mathrm{E}\{\mathbf{B}\mathbf{\Lambda}\mathbf{B}^{\mathrm{H}}\}$ is the Toeplitz matrix, where $\mathbf{B} = [\mathbf{b}(\omega_{1y}), \mathbf{b}(\omega_{2y}), \cdots, \mathbf{b}(\omega_{Ky})]$ with $\mathbf{b}(\omega_{ky}) = [e^{-2j(-2M_1M_2+M_1+M_2)\omega_{ky}}, e^{-2j(-2M_1M_2+M_1+M_2+1)\omega_{ky}}, \cdots, e^{-2j(2M_1M_2-M_1-M_2)\omega_{ky}}]^{\mathrm{T}}$, $\mathbf{\Lambda} = \mathrm{undiag}(\mathbf{R}_{ss}) * \mathrm{undiag}(\mathbf{R}_{ss})^{\mathrm{H}}$, $\mathbf{\Gamma}$ is an $H \times F$ selection matrix, H is the number of elements in the virtual array 1, and $F = (4M_1M_2 - 2M_1 - 2M_2 + 1)$.

C. Estimation of α_k and β_k

We now employ SPA[1] [33] to solve the sparse recovery of (39), obtaining the angle β_k, and subsequently apply the phase retrieval operation to determine the angle α_k.

In order to estimate β_k, SPA is used to recover the Toeplitz matrix $\mathbf{T}$ in (38) as follows

$$\begin{aligned}\min_{x,\mathbf{T}} \quad & x + \mathrm{tr}[\mathbf{\Gamma}\mathbf{T}\mathbf{\Gamma}^{\mathrm{H}}] \\ \text{s.t.} \quad & \begin{bmatrix} x & \bar{\mathbf{r}}_{11}^{\mathrm{H}} \\ \bar{\mathbf{r}}_{11} & \mathbf{\Gamma}\mathbf{T}\mathbf{\Gamma}^{\mathrm{H}} \end{bmatrix} \geq 0.\end{aligned} \tag{39}$$

After obtaining Toeplitz matrix $\mathbf{T}$ through the SDP solver, ω_{ky} can be obtained through root-MUSIC [37]. Then, β_k can be derived via the following expression

$$\beta_k = \arccos\left(\frac{\lambda\omega_{ky}}{-2\pi d}\right). \tag{40}$$

Leveraging the estimated values of β_k, we construct matrix $\overline{\mathbf{B}}_{21}$ based on the formulation of vector $\overline{\mathbf{b}}_{21}(\omega_{ky})$. Then, applying the pseudo-inverse of $\overline{\mathbf{B}}_{21}$ to (37), $\mathrm{undiag}(\mathbf{\Phi}^2\mathbf{R}_{ss})$ is obtained as bellow

$$\mathrm{undiag}(\mathbf{\Phi}^2\mathbf{R}_{ss}) = (\overline{\mathbf{B}}_{21})^{\dagger}\bar{\mathbf{r}}_{12}. \tag{41}$$

According to (26), it follows that ω_{kx} can be derived from the estimated matrix $\mathbf{\Phi}$. Similar to (40), α_k is derived by

$$\alpha_k = \arccos\left(\frac{\mathrm{angle}(\mathrm{undiag}(\mathbf{\Phi}^2\mathbf{R}_{\mathrm{ss}})_k)/2}{-2\pi d/\lambda}\right). \tag{42}$$

It is noteworthy that, owing to the phase retrieval operation, automatic pairing between angles α and β is achieved.

D. Estimation of r_k

In this section, we perform cross-correlation calculation between the data in (11)-(12) and that from the coordinate system's origin in (13), followed by applying the MUSIC algorithm to ultimately derive the range r_k.

First, we concatenate all elements in subarray 1 and subarray 2 to obtain $\mathbf{x}(t)$

$$\mathbf{x}(t) = \begin{bmatrix}\mathbf{x}_1(t) \\ \mathbf{x}_2(t)\end{bmatrix} = \begin{bmatrix}\mathbf{A}_1 \\ \mathbf{A}_2\end{bmatrix}\mathbf{s}(t) + \begin{bmatrix}\mathbf{n}_1(t) \\ \mathbf{n}_2(t)\end{bmatrix} = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \tag{43}$$

where $\mathbf{A}$ is the steering matrix for $\mathbf{x}(t)$, and $\mathbf{n}(t)$ denotes the additive white Gaussian noise vector of the combined array.

Second, the two types of cross-correlation matrices of $\mathbf{x}(t)$ and the sensor set at origin are constructed as follows

$$\begin{aligned}\mathbf{r}_z(\tau) &= \mathrm{E}\{\mathbf{x}(t+\tau)\mathbf{x}_3^{\mathrm{H}}(t)\} \\ &= \mathbf{A}_z\mathbf{r}_s(\tau)\end{aligned} \tag{44}$$

$$\begin{aligned}\mathbf{r}_z(-\tau) &= \mathrm{E}\{\mathbf{x}(t)\mathbf{x}_3^{\mathrm{H}}(t+\tau)\} \\ &= \mathbf{A}_z\mathbf{r}_s(-\tau)\end{aligned} \tag{45}$$

---

[1] SPA performs parameter estimation in the continuous range based on the well-established covariance fitting criterion and convex optimization, which is suitable for uniform linear and sparse redundant arrays across varying snapshot conditions.

where $\mathbf{x}_3 = [x_3(t), x_3(t), \cdots, x_3(t)]^{\mathrm{T}}$ is a $(2M_1 + 2M_2 - 2) \times 1$ vector. Given $\mathbf{r}_s(\tau) = \mathbf{r}_s^*(-\tau)$, $\mathbf{r}_z^*(-\tau)$ can be expressed as follows

$$\begin{aligned} \mathbf{r}_z^*(-\tau) &= (\mathbf{A}_z \mathbf{r}_s(-\tau))^* \\ &= \mathbf{A}_z^* \mathbf{r}_s(\tau). \end{aligned} \tag{46}$$

Then, by cascading $\mathbf{r}_z(\tau)$ and $\mathbf{r}_z^*(-\tau)$, we have

$$\overline{\mathbf{r}}_z(\tau) = \begin{bmatrix} \mathbf{r}_z(\tau) \\ \mathbf{r}_z^*(-\tau) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_z \\ \mathbf{A}_z^* \end{bmatrix} \mathbf{r}_s(\tau) = \overline{\mathbf{A}}_z \mathbf{r}_s(\tau) \tag{47}$$

where $\overline{\mathbf{A}}_z$ is the steering matrix for $\overline{\mathbf{r}}_z(\tau)$. Similar to (28) and (29), with N pseudo snapshots, we have

$$\begin{aligned} \overline{\mathbf{R}}_z &= [\overline{\mathbf{r}}_z(1), \overline{\mathbf{r}}_z(2), \cdots, \overline{\mathbf{r}}_z(N)] \\ &= \overline{\mathbf{A}}_z \mathbf{R}_s(\tau) \end{aligned} \tag{48}$$

where $\mathbf{R}_s(\tau) = [\mathbf{r}_s(1), \mathbf{r}_s(2), \cdots, \mathbf{r}_s(N)]$. By constructing matrix $\overline{\mathbf{A}}_z$ based on estimated 2-D angle parameters and the variable r_k, and then applying the MUSIC algorithm to $\overline{\mathbf{R}}_z$, the estimation of range parameter is achieved.

The proposed algorithm is summarized in TABLE I.

TABLE I

Algorithm Summary

Steps
Step 1: According to (16) and (17), two types of cross-correlation processing are performed on $\mathbf{x}_1(t)$ and $\mathbf{x}_2(t)$, resulting in two cross-correlation matrices $\mathbf{R}_1$ and $\mathbf{R}_2$.
Step 2: Perform covariance calculation on the obtained cross-correlation matrices $\mathbf{R}_1$ and $\mathbf{R}_2$, followed by vectorization, redundancy removal and reordering operations, to derive $\overline{\mathbf{r}}_{11}$ and $\overline{\mathbf{r}}_{12}$.
Step 3: Applying the SPA to matrix $\mathbf{R}$ in (38) results in the computation of ω_{ky}, from which the corresponding α_k and β_k are determined based on (40) and (42).
Step 4: Perform cross-correlation between the data of subarray 1 and subarray 2 with the data of the sensor set at origin to obtain matrix $\overline{\mathbf{R}}_z$, followed by applying the MUSIC method on $\overline{\mathbf{R}}_z$ to derive r_k.

E. Complexity Analysis

In this part, the complexity of the proposed algorithm and the comparison algorithms is analyzed. Complexity analysis involves considering following aspects: 1) the construction of the virtual array; 2) computation of the covariance matrix; 3) Eigenvalue Decomposition (EVD); 4) spectral peak search; 5) the optimization process. Here, define the search interval of α as $\Delta\theta_\alpha$, the search interval of β as $\Delta\theta_\beta$, and the search interval of r as Δr. Let $Ran = 2D^2/\lambda - 0.62\sqrt{D^3/\lambda}$ represents the search range, where D represents array aperture, and DoF represents the degrees of freedom of the virtual array. L represents the number of snapshots, and M_x and M_y represent the number of elements along the x and y coordinates in each algorithm. Then, the complexity of all algorithms is shown in TABLE II.

TABLE II

Algorithm Complexity

Algorithm	Complexity
Proposed	$O\{2(M_1+M_2-1)^2N + (4(M_1+M_2-1))^2N + (4(M_1+M_2-1))^3 + K(4(M_1+M_2-1))^2 Ran/\Delta r + K^2(DoF)^{2.5} + (DoF)^3 + (4(M_1+M_2-1))^3\}$
TSMUSIC [39]	$O\{9(M_x^2+M_y^2)L + 9((M_x-1)/2+1)^2L + M_x^3 + M_y^2 + 9((M_y-1)/2+1)^2L + ((M_x-1)/2+1)^3 + ((M_y-1)/2+1)^3 + \pi M_x^2/\Delta\theta_\alpha + \pi M_y^2/\Delta\theta_\beta + M_y^2 Ran/\Delta r\}$
Challa [38]	$O\{9((M_x-1)^2 + (M_y-1)^2)(L-N)N + 2N((M_x-1)^2 + (M_y-1)^2) + (M_x-1)^6 + (M_y-1)^6 + K^3\}$
Wu [28]	$O\{9M_xM_yL + M_xM_y^2 + K^3 + L(M_x+M_y)^2 T Ran/\Delta r\}$
Chen [32]	$O\{9(M_x^2+M_y^2)(L-N)N + LM_xDoF + (5DoF)^2K + K^2(5DoF)^{2.5} + 2(M_x+M_y)(L-N)N + (2(M_x+M_y))^3 + K(2(M_x+M_y))^2 Ran/\Delta r + (2(M_x+M_y))^2N\}$

F. Discussion

1) Chen's method [32] and Challa's method [38], along with the proposed method all belong to the class of spatial-temporal based NF localization methods. Here we provide the advantages and disadvantages of these three methods. On the one hand, although insensitive to noise, both the compared methods in [32] and [38] resort to fourth-order cumulants to eliminate the range-related quadratic term, while the proposed exploits second-order statistics, which incurs a lower computational complexity to realize 3-D NF localization given a significantly increased number of snapshots, as illustrated in Sec. V-E. On the other hand, under the same number of physical elements, the parameter estimation performance of the proposed algorithm is comparable to that of Chen's method; however, it significantly outperforms Challa's algorithm, offering the ability to perform underdetermined mixed source parameter estimation.

2) In this paper, we employ a sparse recovery method named SPA [33] to estimate the angle β, the angle α is achieved by a phase retrieval operation, and the range r using the 1-D MUSIC method. The reason is that in (36) and (37), the vectorized data $\overline{\mathbf{r}}_{11}$ is a single snapshot, behaving like a coherent data model that cannot directly use the MUSIC algorithm without de-correlation operation. On the contrary, SPA is robust to the coherent signal model, and adopted for the estimate of angle β. For the estimate of angle α, we use a phase retrieval operation, which is for the purpose of simplicity and convenience. To estimate the range r, we resort to the original multiple numbers of data of the TPSC array along with its conjugate data, and MUSIC is used which not only offers faster computation but also provides higher precision in the estimated parameters. However, it is possible to solve for r using the l_1 norm in sparse recovery, which may achieve good estimation performance by reasonable selection of experiential regularization parameters.

3) Subspace or sparse based direction-finding methods based on covariance matrices require those matrices to be calculated in a statistical sense. Considering that the actual number of received data snapshots is finite, in this paper

we use maximum likelihood estimation to obtain the second-order statistics based covariance matrix $\mathbf{R}$ with (38) in the time averaged sense. A higher number of snapshots enhance the precision of the estimated covariance matrix, thereby improving the accuracy of subsequent parameter estimation, while a lower number of snapshots may result in less accurate parameter estimates.

IV. ANALYSIS OF TPSC ARRAY PERFORMANCE

In order to show the enhanced parameter estimation performance using the designed TPSC array, we will analyze some properties of the TPSC array.

A. Properties of Virtual Array

According to Sec. III-B, two virtual arrays are generated. We only address virtual array 1 in this part. Therefore, any further mention of virtual array in this section will be in reference to virtual array 1.

Proposition 1: *The virtual array is symmetric about the origin, with range* $[-(2M_1M_2-M_1-M_2),(2M_1M_2-M_1-M_2)]$.

Proof: Based on the array model in Fig. 1, since subarray 1 and subarray 2 exhibit centrosymmetry about the origin, taking subarray 1 as an example, subarray 1 consists of two ULAs, i.e.,

$$\mathbb{S}_1=\{m_1M_2,-(M_1-1)\leqslant m_1\leqslant 0\} \tag{49}$$

$$\mathbb{S}_2=\{m_2M_1,0\leqslant m_2\leqslant (M_2-1)\} \tag{50}$$

where $\mathbb{S}_1$ and $\mathbb{S}_2$ respectively represent the y-coordinate sets of all elements in ULA 1 and ULA 2 of subarray 1, $M_1>M_2$. Therefore, the y-coordinate values set of all elements in subarray 1 can be expressed as

$$\mathbb{S}=\mathbb{S}_1\cup\mathbb{S}_2. \tag{51}$$

Here, we take into account the scope of virtual arrays across various combination scenarios. Assume that $l_c=s_1-s_2$, where l_c denotes the y-coordinate values of the elements in the virtual array, $s_1,s_2\in\mathbb{S}$. The various cases can be summarized as follows.

1) $s_1,s_2\in\mathbb{S}_1$: define $s_1=m_{1a}M_2$ and $s_2=m_{1b}M_2$, where $m_{1a},m_{1b}\in[-(M_1-1),0]$, and it can be deduced that

$$l_c=s_1-s_2=(m_{1a}-m_{1b})M_2. \tag{52}$$

Evidently, the range of $m_{1a}-m_{1b}$ is $[-(M_1-1),(M_1-1)]$, and thereby the virtual array is symmetric about the origin. Let $\mathbb{L}_1$ denotes the set of y-coordinate values of the virtual array elements in this case. The range of $\mathbb{L}_1$ is $[-(M_1-1)M_2,(M_1-1)M_2]$.

2) $s_1\in\mathbb{S}_1$, $s_2\in\mathbb{S}_2$: we have

$$l_c=s_1-s_2=m_1M_2-m_2M_1. \tag{53}$$

3) $s_1\in\mathbb{S}_2$, $s_2\in\mathbb{S}_1$: we have

$$l_c=s_1-s_2=m_2M_1-m_1M_2. \tag{54}$$

By analyzing the expressions derived from (53) and (54), it becomes evident that the $\mathbb{L}_2$ formed by these two cases is symmetric about the origin. The range of $\mathbb{L}_2$ is $[-(2M_1M_2-M_1-M_2),(2M_1M_2-M_1-M_2)]$.

4) $s_1,s_2\in\mathbb{S}_2$: define $s_1=m_{2a}M_1$ and $s_2=m_{2b}M_1$, where $m_{2a},m_{2b}\in[0,(M_2-1)]$, and we have

$$l_c=s_1-s_2=(m_{2a}-m_{2b})M_1. \tag{55}$$

In this case, the expression for the y-coordinate values of the virtual array elements is given by $(m_{2a}-m_{2b})M_1$. Given that $m_{2a}-m_{2b}$ is symmetric about the origin, and M_1 is a constant, the virtual array is symmetric about the origin. Let $\mathbb{L}_3$ denotes the set of y-coordinate values of the virtual array elements in this case. The range of $\mathbb{L}_3$ is $[-(M_2-1)M_1,(M_2-1)M_1]$.

To summarize, the set of virtual array elements is $\mathbb{L}=\mathbb{L}_1\cup\mathbb{L}_2\cup\mathbb{L}_3$. Thus, the virtual array retains symmetry about the origin across all cases, and the range of the virtual array is $[-(2M_1M_2-M_1-M_2),(2M_1M_2-M_1-M_2)]$.

Proposition 2: *Redundant elements*[2] *are present within the virtual array.*

Proof: The collection of y-coordinate values of actual array elements is denoted as $\mathbb{S}$. The crux of the issue lies in demonstrating whether there are any redundant elements derived from the difference between s_{11} and s_{12}, where $s_{11},s_{12}\in\mathbb{S}$.

The y-coordinate values of the array elements in the resultant set are

$$l_c=s_{11}-s_{12} \tag{56}$$

where l_c represents the y-coordinate value of the element in the resultant set.

It can be inferred that there exists a case where $s_{11}=-(M_1-1)M_2$ and $s_{12}=-(M_1-1)M_2$, resulting in $l_c=0$. Similarly, when $s_{11}=(M_2-1)M_1$ and $s_{12}=(M_2-1)M_1$, $l_c=0$ as well. Consequently, the combination of two sets of $\mathbb{S}$ introduces redundancy.

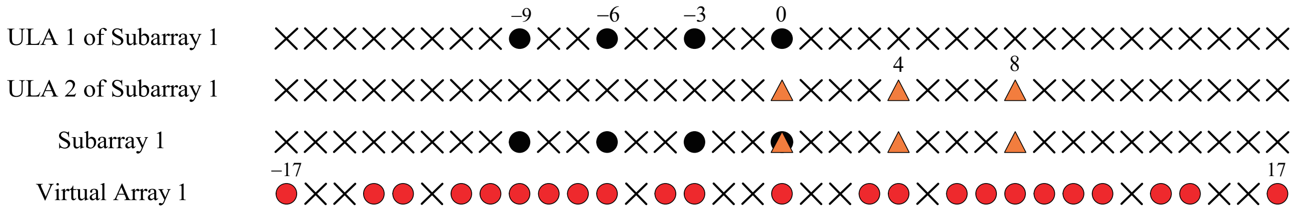


Fig. 3. An array example with $Q=13$.

---

[2] The term 'redundant elements' refers to cases where, in the difference co-array, multiple elements may occupy the same coordinate, resulting in unnecessary duplication.

B. Design of M_1 and M_2

Based on the model described in Sec. II, the total number of elements in the TPSC array is $Q = 2(M_1 + M_2) - 1$. According to Proposition 1, the range of the virtual array is $[-(2M_1M_2 - M_1 - M_2), (2M_1M_2 - M_1 - M_2)]$, and $\xi = (2M_1M_2 - M_1 - M_2)$. Therefore, when the total number of elements in the TPSC array is fixed, changing M_1 and M_2 can alter the aperture of the virtual array. Through the following analysis, we determine the configuration of M_1 and M_2 for maximizing the aperture of the virtual array.

Since the total number of elements in the TPSC array is a constant $Q = 2(M_1 + M_2) - 1$, $w = M_1 + M_2$ is also a constant. Substituting w into the expression of ξ, we have

$$\begin{aligned}\xi &= 2M_1(w - M_1) - M_1 - (w - M_1)\\ &= -2(M_1 - \frac{w}{2})^2 + \frac{w^2}{2} - w.\end{aligned} \tag{57}$$

The quadratic function ξ reaches its maximum value at $M_1 = w/2$, but due to the unknown parity of w, further discussion is necessary.

1) If w is odd, one of M_1 and M_2 is odd while the other is even, then the maximum value of ξ is attained at $M_1 = (w + 1)/2$, with $M_2 = (w - 1)/2$. Utilizing an expression involving Q to represent M_1 and M_2 leads to the derivation of $M_1 = (Q + 3)/4$ and $M_2 = (Q - 1)/4$. In this case, the aperture of the virtual array is

$$D = 2\xi = \frac{Q^2 - 2Q - 7}{4}. \tag{58}$$

2) If w is even and $w/2$ is odd, and both M_1 and M_2 are odd, it can be inferred that the maximum value of ξ is attained at $M_1 = (w + 4)/2$, with $M_2 = (w - 4)/2$. Utilizing an expression involving Q to represent M_1 and M_2 leads to the derivation of $M_1 = (Q + 9)/4$ and $M_2 = (Q - 7)/4$. In this case, the aperture of the virtual array is as follows

$$D = 2\xi = \frac{Q^2 - 2Q - 67}{4}. \tag{59}$$

3) If both w and $w/2$ are even, and both M_1 and M_2 are odd, then the maximum value of ξ is attained at $M_1 = (w + 2)/2$, with $M_2 = (w - 2)/2$. Utilizing an expression involving Q to represent M_1 and M_2 leads to the derivation of $M_1 = (Q + 5)/4$ and $M_2 = (Q - 3)/4$. In this case, the aperture of the virtual array is

$$D = 2\xi = \frac{Q^2 - 2Q - 19}{4}. \tag{60}$$

The recommended optimal parameter configuration is detailed in TABLE III. To illustrate the idea of Sec. IV-B, an example is provided in Fig. 3, consisting of a total of $Q = 13$ elements with $M_1 = 4$ and $M_2 = 3$.

TABLE III
OPTIMAL PARAMETER CONFIGURATION FOR TPSC ARRAY

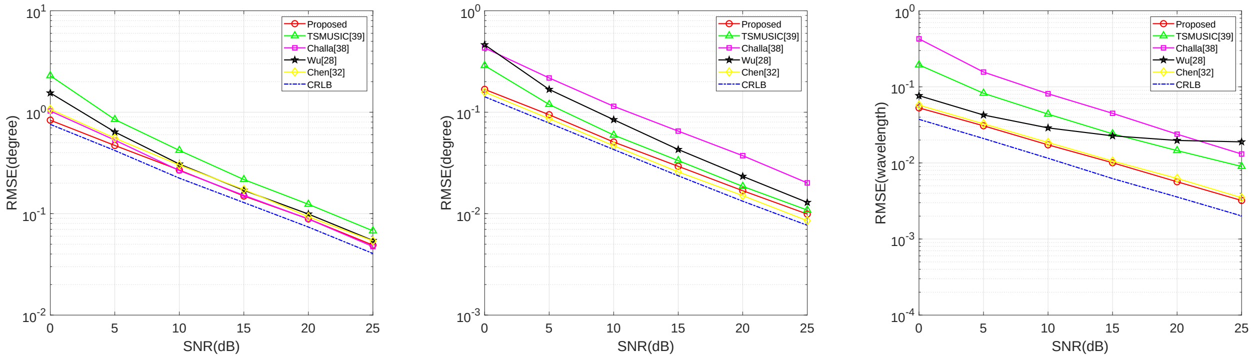
Number of physical elements	$M_1 + M_2$	$(M_1 + M_2)/2$	M_1	M_2	Aperture of virtual array
$Q = 2(M_1 + M_2) - 1$	Even	Even	$\frac{Q+5}{4}$	$\frac{Q-3}{4}$	$\frac{Q^2-2Q-19}{4}$
		Odd	$\frac{Q+9}{4}$	$\frac{Q-7}{4}$	$\frac{Q^2-2Q-67}{4}$
	Odd		$\frac{Q+3}{4}$	$\frac{Q-1}{4}$	$\frac{Q^2-2Q-7}{4}$

V. SIMULATION RESULTS

In this section, performance of the proposed algorithm based on the TPSC array is demonstrated and compared with Wu's algorithm [28], Challa's algorithm [38], TSMUSIC [39], and Chen's algorithm [32] as well as the Cramér-Rao lower bound (CRLB) derived in [32]. The impinging non-Gaussian source signals are modelled as $e^{j\varphi_t}$, where φ_t is uniformly distributed in $[0, 2\pi]$. In the simulations, all sources' power is set to 1. The root mean square error (RMSE) for parameter estimation is defined as follows

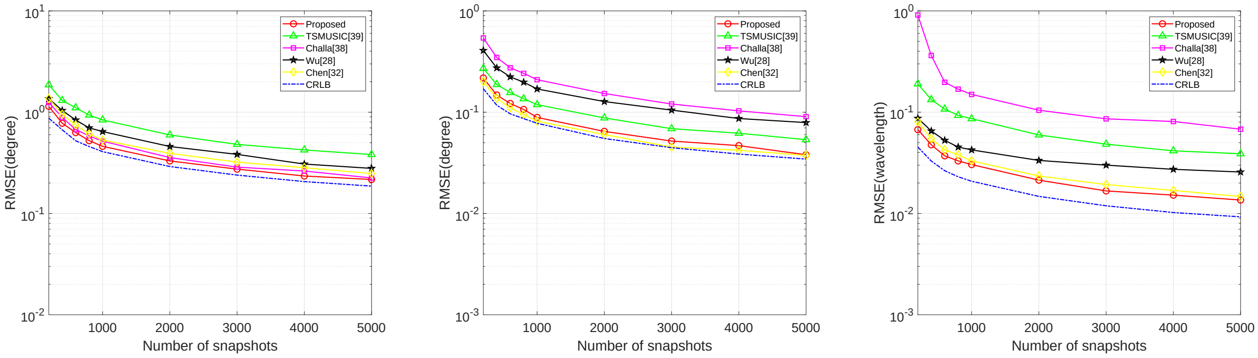
$$RMSE = \sqrt{\frac{1}{M_cK}\sum_{i=1}^{M_c}\sum_{j=1}^{K}\left(\hat{x}_j^{(i)} - x_j\right)^2} \tag{61}$$

where M_c is the number of Monte Carlo runs, K is the number of sources, $\hat{x}_j^{(i)}$ denotes the estimation of the jth signal source during the ith run, and x_j is the actual value of the jth NF source.



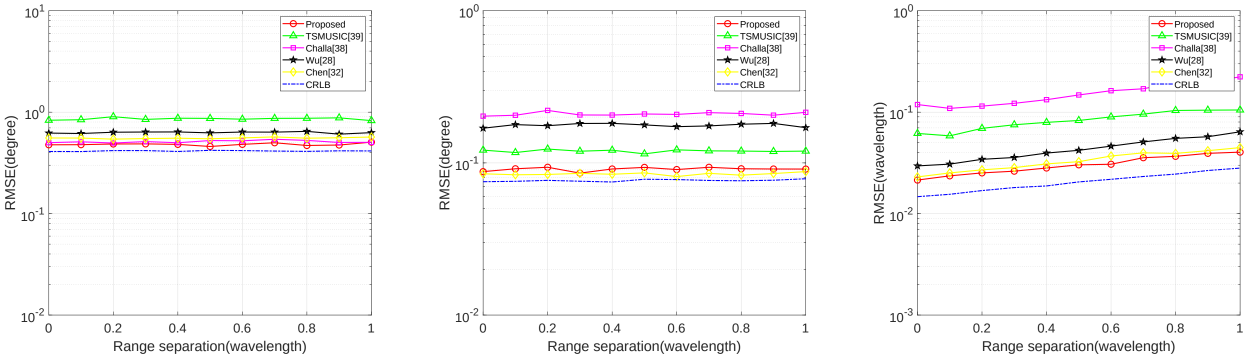
(a) α's RMSE (b) β's RMSE (c) r's RMSE

Fig. 4. RMSE versus SNR.



(a) α's RMSE (b) β's RMSE (c) r's RMSE

Fig. 5. RMSE versus number of snapshots.



(a) α's RMSE (b) β's RMSE (c) r's RMSE

Fig. 6. RMSE versus range separation.

A. RMSE versus SNR

Fig. 4 shows the RMSE curve versus SNR for different algorithms. The 3-D parameters of the two NF sources are $(55°, 70°, 1.8\lambda)$ and $(120°, 117°, 2.3\lambda)$. The total number of elements of all algorithms is set to 9. TSMUSIC and Wu's algorithm both adopt a symmetric ULA with 3 elements on the x-axis and 7 elements on the y-axis, and Chen's algorithm sets 7 elements on the y-axis with $M_1 = 3$ and $M_2 = 2$, and 3 elements on the x-axis, while for the proposed, we set $M_1 = 3$ and $M_2 = 2$. Given that Challa's algorithm requires an even number of elements along both axes, the simulation has been configured with 4 elements on the x-axis and 6 elements on the y-axis. The total number of snapshots is 1000, while the number of pseudo snapshots of the proposed algorithm, Chen's algorithm, and Challa's algorithm is set to 50. The SNR ranges from 0 to 25dB. The number of Monte Carlo trials is 500.

As illustrated in Fig. 4, the proposed algorithm significantly outperforms the comparison algorithms for angle α, owing to the greater number of virtual elements generated along the x-axis. While for angle β, the proposed algorithm is only slightly behind Chen's algorithm due to its virtual array aperture along the y-axis being $(4M_1M_2 - 2M_1 - 2M_2)$, marginally smaller than the virtual array aperture $(4M_1M_2 - 4M_2)$ along the y-axis produced by Chen's algorithm. For the range parameter, the proposed algorithm is better than Wu's algorithm, Challa's algorithm and TSMUSIC, and similar to Chen's algorithm, as both the proposed algorithm and Chen's algorithm employ the 1-D MUSIC algorithm to determine the range parameter, leading to comparable estimation performance.

B. RMSE versus Number of Snapshots

All configurations remain the same as those in Sec. V-A, except for the number of snapshots and SNR. The SNR is set to 5dB, while the number of snapshots varies from 200 to 5000. As depicted in Fig. 5, the proposed algorithm demonstrates superior result for angle α; however, its angle β result is slightly worse than Chen's algorithm, while for range parameter it closely aligns with Chen's algorithm. Notably, the RMSE of the 3-D parameters decreases with an increasing number of snapshots, eventually reaching a flat state.

C. RMSE versus Range Separation

In this part, we investigate the impact of range separation on the performance. The parameters are the same as in Sec. V-A except for the range parameter and SNR. The SNR is set to 5dB, and the first range parameter is set as 1.8λ. Initially, the second range parameter is identical to the first, and the range separation $\Delta\lambda$ of the second source varies from 0 to 1λ. As shown in Fig. 6, the estimation performance for 2-D angle parameters is minimally influenced by range separation, while the range parameter estimation result gets worse with increasing range separation. The reason is that, with the increase of range separation, the second range gets closer to that of an FF source, affecting the accuracy in range estimation.

D. Underdetermined Case

In this part, the underdetermined case is considered with 20 NF signals incident on the array, characterized by $(51^\circ, 112^\circ, 3.7\lambda)$, $(66^\circ, 77^\circ, 6.7\lambda)$, $(56^\circ, 117^\circ, 7.1\lambda)$, $(96^\circ, 42^\circ, 5.6\lambda)$, $(113^\circ, 90^\circ, 7.9\lambda)$, $(83^\circ, 139^\circ, 3.0\lambda)$, $(60^\circ, 29^\circ, 3.1\lambda)$, $(105^\circ, 154^\circ, 1.5\lambda)$, $(147^\circ, 14^\circ, 5.7\lambda)$, $(61^\circ, 63^\circ, 2.6\lambda)$, $(134^\circ, 48^\circ, 3.3\lambda)$, $(78^\circ, 59^\circ, 1.2\lambda)$, $(109^\circ, 68^\circ, 1.9\lambda)$, $(118^\circ, 107^\circ, 7.6\lambda)$, $(47^\circ, 92^\circ, 4.1\lambda)$, $(78^\circ, 94^\circ, 2.6\lambda)$, $(137^\circ, 134^\circ, 1.6\lambda)$, $(141^\circ, 65^\circ, 6.7\lambda)$, $(35^\circ, 125^\circ, 3.9\lambda)$, $(37^\circ, 156^\circ, 6.0\lambda)$. The TPSC array comprises a total of 17 elements with $M_1 = 5$ and $M_2 = 4$. The SNR is set to 30dB. The number of snapshots is 5000 with the number of pseudo snapshots being 100. The number of Monte Carlo trials is 100. The results are shown in Fig. 7. It can be observed that all 20 NF sources have been distinguished successfully, showing that the proposed algorithm remains effective in underdetermined case.

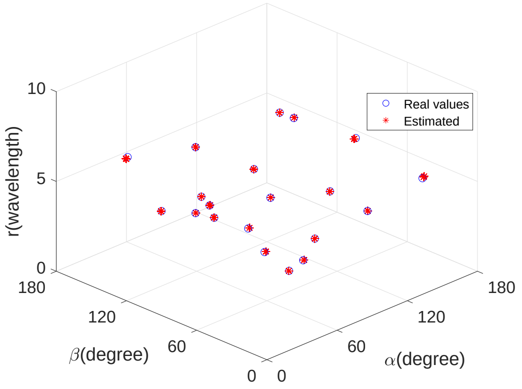


Fig. 7. Locations of the 20 estimated NF sources.

E. Runtime Comparison

In this part, the runtime of each algorithm is compared. The simulation parameters are set the same as in Sec. V-B, except for the number of snapshots which varies from 1000 to 18600. The result is shown in Fig. 8. It can be observed that the runtime of the proposed algorithm is lower than Wu's algorithm and Chen's algorithm. When the number of snapshots is under 6000, the proposed algorithm takes longer time to run than TSMUSIC; however, as the number of snapshots increases beyond 6000, it achieves better runtime performance. This is because TSMUSIC avoids the time-consuming SPA used in our method, making it faster when the number of snapshots is small. However, as the number of snapshots increases, its reliance on fourth-order cumulants slows TSMUSIC down, causing it to perform worse than the proposed algorithm. Although Challa's algorithm is faster for a smaller number of snapshots, its runtime performance declines as the number of snapshots increases significantly, due to higher complexity of the fourth-order cumulants calculation, which results in longer runtime and ultimately approaches the runtime of the proposed algorithm.

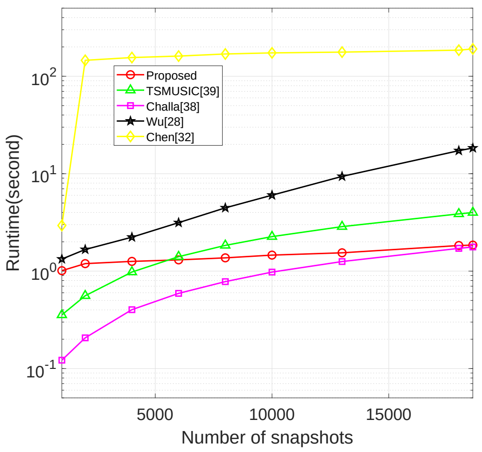


Fig. 8. Runtime comparison.

F. Mixed Source Scenario

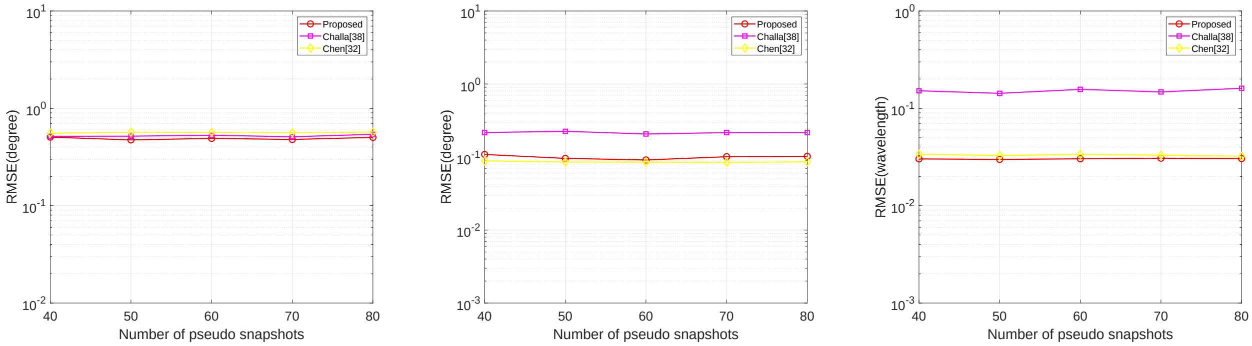
Considering the scenario with mixed NF and FF signals, where all other parameters are the same as those in Sec. V-D except for the signal type. There are 12 NF signals and 5 FF signals, with their parameters given by $(149^\circ, 149^\circ, 0.4\lambda)$, $(139^\circ, 139^\circ, 0.8\lambda)$, $(130^\circ, 130^\circ, 1.2\lambda)$, $(123^\circ, 123^\circ, 1.6\lambda)$, $(115^\circ, 115^\circ, 2.0\lambda)$, $(109^\circ, 109^\circ, 2.4\lambda)$, $(102^\circ, 102^\circ, 2.8\lambda)$, $(96^\circ, 96^\circ, 3.2\lambda)$, $(90^\circ, 90^\circ, 3.6\lambda)$, $(84^\circ, 84^\circ, 4.0\lambda)$, $(78^\circ, 78^\circ, 4.4\lambda)$, $(71^\circ, 71^\circ, 4.8\lambda)$, $(31^\circ, 31^\circ)$, $(41^\circ, 41^\circ)$, $(50^\circ, 50^\circ)$, $(57^\circ, 57^\circ)$, and $(65^\circ, 65^\circ)$. The results shown in Fig. 9(a) indicate that the angle parameters of all signals are accurately estimated and matched, and all NF signals are localized successfully in Fig. 9(b), thereby demonstrating the proposed algorithm's viability in the mixed-source scenario.

G. RMSE versus Number of pseudo Snapshots

Finally, we investigate the impact of the number of pseudo snapshots on the performance, where we only consider the delay-based localization methods [32, 38]. All configurations remain the same as those in Sec. V-A, except for the number of pseudo snapshots and SNR. The SNR is set to 5dB, while the number of pseudo snapshots varies from 40 to 80. As shown in Fig. 10, the parameter estimation performance by all the compared methods is barely affected by the varying number of pseudo snapshots.

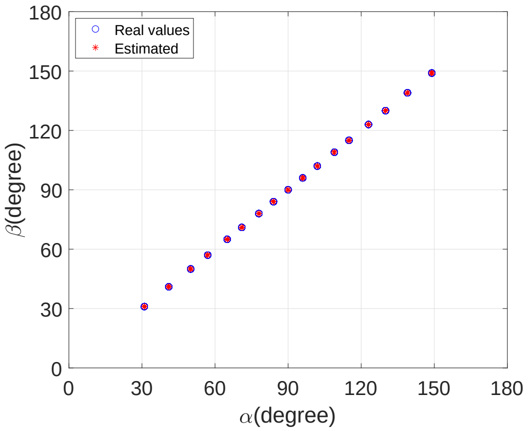
VI. CONCLUSIONS

In this paper, a 3-D parameter estimation algorithm has been proposed for NF sources based on a newly constructed TPSC

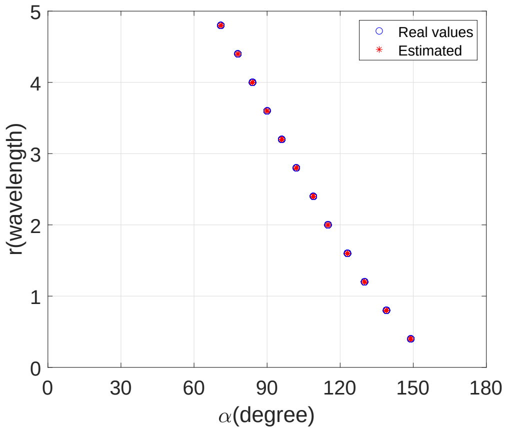


(a) α's RMSE (b) β's RMSE (c) r's RMSE

Fig. 10. RMSE versus number of pseudo snapshots.



(a) Estimation results of 2-D angles.



(b) Estimation results of angle α and range.

Fig. 9. Parameters estimation in mixed-source scenarios.

array. Firstly, leveraging the central symmetry of subarray 1 and subarray 2, two cross-correlation matrices are constructed. Subsequently, by subjecting these matrices to covariance calculation and vectorization, it facilitates the virtual FF transformation of array data, and expanding the virtual array aperture. Following this, 2-D angle estimation is achieved using SPA and a phase retrieval operation. Finally, the range parameter is estimated utilizing the 1-D MUSIC algorithm. Simulation results have demonstrated that the proposed algorithm significantly outperforms the comparison algorithms in terms of estimation accuracy and it can also effectively deal with underdetermined and mixed-source scenarios.

REFERENCES

[1] W. Liu, M. Haardt, M. S. Greco, C. F. Mecklenbruker, and P. Willett, "Twenty-five years of sensor array and multichannel signal processing: A review of progress to date and potential research directions," *IEEE Signal Processing Magazine*, vol. 40, no. 4, pp. 80–91, 2023.

[2] F. Wen, H. Wang, G. Gui, H. Sari, and F. Adachi, "Polarized intelligent reflecting surface aided 2D-DOA estimation for NLoS sources," *IEEE Transactions on Wireless Communications*, vol. 23, no. 7, pp. 8085–8098, 2024.

[3] L. Wan, K. Liu, Y.-C. Liang, and T. Zhu, "DOA and polarization estimation for non-circular signals in 3-D millimeter wave polarized massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 5, pp. 3152–3167, 2021.

[4] X. Wu, Z. Yang, Z. Wei, and Z. Xu, "Direction-of-arrival estimation for constant modulus signals using a structured matrix recovery technique," *IEEE Transactions on Wireless Communications*, vol. 23, no. 4, pp. 3117–3130, 2024.

[5] J. He, Y. Wang, T. Shu, and T.-K. Truong, "Polarization, angle, and delay estimation for tri-polarized systems in multipath environments," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 5828–5841, 2022.

[6] H. Chen, J. Fang, W. Wang, W. Liu, Y. Tian, Q. Wang, and G. Wang, "Near-field target localization for EMVS-MIMO radar with arbitrary configuration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 4, pp. 5406–5417, 2024.

[7] S. Yang, X. Chen, Y. Xiu, W. Lyu, Z. Zhang, and C. Yuen, "Performance bounds for near-field localization with widely-spaced multi-subarray mmwave/thz MIMO," *IEEE Transactions on Wireless Communications*, vol. 23, no. 9, pp. 10 757–10 772, 2024.

[8] J. He, L. Li, T. Shu, and T.-K. Truong, "Mixed near-field and far-field source localization based on exact spatial propagation geometry," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3540–3551, 2021.

[9] X. Wu, C. You, J. Li, and Y. Zhang, "Near-field beam training: Joint angle and range estimation with DFT codebook," *IEEE Transactions on Wireless Communications*, vol. 23, no. 9, pp. 11 890–11 903, 2024.

[10] C. Weng, X. Guo, and Y. Wang, "Near-field beam training with hierarchical codebook: Two-stage learning-based approach," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 14 003–14 008, 2024.

[11] X. Zhang, H. Zhang, and Y. C. Eldar, "Near-field sparse channel representation and estimation in 6G wireless communications," *IEEE Transactions on Communications*, vol. 72, no. 1, pp. 450–464, 2024.

[12] S. Yang, C. Xie, W. Lyu, B. Ning, Z. Zhang, and C. Yuen, "Near-field channel estimation for extremely large-scale reconfigurable intelligent surface (XL-RIS)-aided wideband mmwave systems," *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 6, pp. 1567–1582, 2024.

[13] A. M. Elbir, W. Shi, A. K. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, “Near-field terahertz communications: Model-based and model-free channel estimation,” *IEEE Access*, vol. 11, pp. 36 409–36 420, 2023.

[14] B. Wang, Y. Zhao, and J. Liu, “Mixed-order music algorithm for localization of far-field and near-field sources,” *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 311–314, 2013.

[15] S. Yang, S. Gao, G. Cui, L. Kong, T. Zhang, and J. Yang, “Iterative self-calibration algorithm for near-field source localization,” in *2016 CIE International Conference on Radar (RADAR)*, 2016, pp. 1–6.

[16] Y. Tian, Q. Lian, and H. Xu, “Mixed near-field and far-field source localization utilizing symmetric nested array,” *Digital Signal Processing*, vol. 73, pp. 16–23, 2018.

[17] X. Wu and J. Yan, “A second-order statistics-based mixed sources localization method with symmetric sparse arrays,” *IEEE Communications Letters*, vol. 24, no. 8, pp. 1695–1699, 2020.

[18] X. Su, P. Hu, Z. Liu, T. Liu, B. Peng, and X. Li, “Mixed near-field and far-field source localization based on convolution neural networks via symmetric nested array,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7908–7920, 2021.

[19] Z. Zheng, M. Fu, W.-Q. Wang, and H. C. So, “Symmetric displaced coprime array configurations for mixed near- and far-field source localization,” *IEEE Transactions on Antennas and Propagation*, vol. 69, no. 1, pp. 465–477, 2021.

[20] J. Zhang, Y. Li, J. Qi, and Y. Li, “Symmetric extended nested array for passive localization of mixture near- and far-field sources,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 3, pp. 1244–1248, 2023.

[21] Y. Tian, X. Wang, X. Jin, and Y. Hou, “Near-field source location estimation algorithm based on coprime array,” in *2022 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, 2022, pp. 1–5.

[22] Z. Zheng, Y. Huang, W.-Q. Wang, and H. C. So, “Augmented covariance matrix reconstruction for DOA estimation using difference coarray,” *IEEE Transactions on Signal Processing*, vol. 69, pp. 5345–5358, 2021.

[23] J. He, T. Shu, L. Li, and T.-K. Truong, “Mixed near-field and far-field localization and array calibration with partly calibrated arrays,” *IEEE Transactions on Signal Processing*, vol. 70, pp. 2105–2118, 2022.

[24] C. Zhou, Y. Gu, Z. Shi, and M. Haardt, “Structured nyquist correlation reconstruction for DOA estimation with sparse arrays,” *IEEE Transactions on Signal Processing*, vol. 71, pp. 1849–1862, 2023.

[25] X. Dong, J. Zhao, M. Sun, X. Zhang, and Y. Wang, “A modified δ-generalized labeled multi-bernoulli filtering for multi-source DOA tracking with coprime array,” *IEEE Transactions on Wireless Communications*, vol. 22, no. 12, pp. 9424–9437, 2023.

[26] Z. Tan, Y. C. Eldar, and A. Nehorai, “Direction of arrival estimation using co-prime arrays: A super resolution viewpoint,” *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5565–5576, 2014.

[27] R. Cohen and Y. C. Eldar, “Sparse array design via fractal geometries,” *IEEE Transactions on Signal Processing*, vol. 68, pp. 4797–4812, 2020.

[28] X. Wu and J. Yan, “3-D mixed far-field and near-field sources localization with cross array,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6833–6837, 2020.

[29] Z. Ma, D. He, X. Chen, and W. Yu, “Localization of 3-D near-field sources based on the joint phase interferometer and music algorithm,” in *2020 IEEE Globecom Workshops (GC Wkshps*, 2020, pp. 1–6.

[30] X. Wu, J. Wang, Y. Liu, and J. Li, “A near-field source localization method for uniform/sparse centrally symmetric rectangular arrays,” in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 13 111–13 115.

[31] J. Fang, J. Liu, H. Chen, W. Liu, Y. Tian, and G. Wang, “Three-dimensional spatial-temporal near-field passive localization based on an exact spatial propagation model,” in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 8516–8520.

[32] H. Chen, Z. Yi, Z. Jiang, W. Liu, Y. Tian, Q. Wang, and G. Wang, “Spatial-temporal-based underdetermined near-field 3-D localization employing a nonuniform cross array,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 18, no. 4, pp. 561–571, 2024.

[33] Z. Yang, L. Xie, and C. Zhang, “A discretization-free sparse and parametric approach for linear array signal processing,” *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 4959–4973, 2014.

[34] Y. Cao, T. Lv, Z. Lin, P. Huang, and F. Lin, “Complex resnet aided DOA estimation for near-field MIMO systems,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11 139–11 151, 2020.

[35] Z. Jiang, H. Chen, W. Liu, Y. Tian, and G. Wang, “Conjugate augmented spatial-temporal near-field sources localization with cross array,” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022, pp. 5113–5113.

[36] H. Chen, W. Wang, and W. Liu, “Joint DOA, range, and polarization estimation for rectilinear sources with a COLD array,” *IEEE Wireless Communications Letters*, vol. 8, no. 5, pp. 1398–1401, Oct 2019.

[37] B. Rao and K. Hari, “Performance analysis of Root-Music,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1939–1949, 1989.

[38] R. N. Challa and S. Shamsunder, “Passive near-field localization of multiple non-gaussian sources in 3-D using cumulants,” *Signal Processing*, vol. 65, no. 1, pp. 39–53, 1998.

[39] J. Liang and D. Liu, “Passive localization of mixed near-field and far-field sources using two-stage MUSIC algorithm,” *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 108–120, 2010.