SubTTD: DOA Estimation via Sub-Nyquist Tensor Train Decomposition

Hang Zheng, *Student Member, IEEE*, Chengwei Zhou, *Member, IEEE*, Zhiguo Shi, *Senior Member, IEEE*, and André L. F. de Almeida, *Senior Member, IEEE*

***Abstract*—Conventional tensor direction-of-arrival (DOA) estimation methods for sparse arrays apply canonical polyadic decomposition (CPD) to the high-order coarray covariance tensor for retrieving angle information. However, due to the low convergence rate of CPD-based algorithms for high-order tensors, these methods suffer from a high computation cost. To address this issue, a sub-Nyquist tensor train decomposition (SubTTD)-based DOA estimation method is proposed for a three-dimensional (3-D) sparse array, where an augmented virtual array is derived from the sub-Nyquist tensor statistics. To reduce computational complexity of processing the 6-D coarray covariance tensor, the proposed SubTTD model efficiently decomposes it into a train of head matrix, 3-D core tensors, and tail matrix. Based on that, a core tensor decomposition and a change-of-basis transformation for the head matrix are designed to retrieve canonical polyadic factors of the coarray covariance tensor for DOA estimation. The computational efficiency of the proposed method is theoretically analyzed, and its effectiveness is verified via simulations.**

***Keywords*— Coarray covariance tensor, DOA estimation, sparse array, sub-Nyquist tensor train decomposition.**

I. Introduction

The deployment of multi-dimensional sparse arrays for super-resolution and high-accuracy direction-of-arrival (DOA) estimation with a reduced system overload has gained increasing popularity [1–5]. To exploit the structural characteristics embedded in multi-dimensional signals sampled at the sub-Nyquist rate, the emerging tensor-based methods have modeled these signals as sub-Nyquist tensors, and then derived augmented virtual arrays for coarray tensor processing [6–9]. Nevertheless, the typical canonical polyadic decomposition (CPD) operation on the corresponding high-order coarray covariance tensor has a low convergence rate, resulting in a high computational complexity for DOA estimation. Thus, devising a computationally efficient coarray tensor-based DOA estimation method is an urgent and important task.

To efficiently process a high-order tensor, the emerging tensor train decomposition model transforms it into a train of interconnected low-order core tensors with a reduced dimensionality [10]. These core tensors maintain the spatial information of the original high-order tensor, while facilitating signal processing tasks with a minimum information loss. Hence, tensor train decomposition has been utilized for harmonic retrieval [11], image recovery [12], hyperspectral image denoising [13], channel estimation [14], and video tracking [15], etc. However, the existing tensor train decomposition techniques have not considered the DOA estimation problem. Moreover, in the context of sparse array processing, the canonical polyadic (CP) factors of the coarray covariance tensor are required for DOA estimation, which however, are not explicitly included in the conventional tensor train decomposition model. Therefore, developing a suitable tensor train decomposition model for sparse array DOA estimation remains challenging.

In this letter, a sub-Nyquist tensor train decomposition (SubTTD)-based DOA estimation method is proposed with a high computational efficiency. First, an augmented virtual cubic array is derived from the sub-Nyquist tensor statistics of a three-dimensional (3-D) sparse array. Then, a SubTTD model is formulated to decompose the corresponding 6-D coarray covariance tensor into a train of head matrix, 3-D core tensors, and tail matrix. Based on the algebraic relevance between the SubTTD and CPD structures, the interconnected SubTTD head matrix and core tensors are converted into the CP factors of the coarray covariance tensor, leading to an efficient DOA estimation. It is proved that the computational complexity of the proposed method is reduced by several times compared to the CPD-based method. Simulation results verify that the proposed method is computationally faster than the competing methods with a better estimation accuracy.

II. Sub-Nyquist Tensor Model for Sparse Array

Among diverse types of sparse arrays [16–19], the Nested Coprime Array with Compressed Inter-element Spacing (NCACIS) [16] can derive a contiguous virtual array. Following its framework, as shown in Fig. 1, we consider the cubic version of NCACIS[1], namely, cubic NCACIS, which consists of a dense uniform cubic array (UCA) $\mathbb{M}$ with $M_x \times M_y \times M_z$

[1]The proposed sub-Nyquist tensor model can be generalized to other typical sparse arrays. For those partially augmentable arrays, such as coprime array, we can either extract the contiguous part of the corresponding discontinuous virtual array [20, 21] or perform interpolation [22, 23] for coarray processing.

This work was supported in part by the National Key R&D Program of China (No. 2018YFE0126300), the National Natural Science Foundation of China (No. 61901413, U21A20456), the Zhejiang University Education Foundation Qizhen Scholar Foundation, and the 5G Open Laboratory of Hangzhou Future Sci-Tech City. (*Corresponding author: Chengwei Zhou*)

H. Zheng is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: hangzheng@zju.edu.cn).

C. Zhou is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China, and also with the Key Laboratory of Collaborative Sensing and Autonomous Unmanned Systems of Zhejiang Province, Hangzhou 310015, China (e-mail: zhouchw@zju.edu.cn).

Z. Shi is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China, and also with the International Joint Innovation Center, Zhejiang University, Haining 314400, China (e-mail: shizg@zju.edu.cn).

A. L. F. de Almeida is with the Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza 60020-181, Brazil (e-mail: andre@gtel.ufc.br).

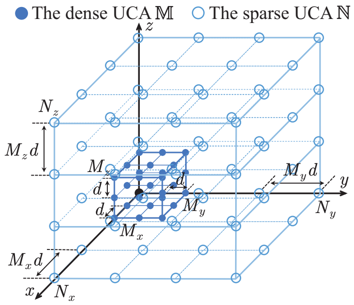

Fig. 1. The geometry of the considered cubic NCACIS.

sensors and a sparse UCA $\mathbb{N}$ with $N_x \times N_y \times N_z$ sensors. Here, (M_x, N_x), (M_y, N_y) and (M_z, N_z) are pairs of coprime integers. The inter-element spacing for the dense UCA $\mathbb{M}$ is a half-wavelength d, whereas the inter-element spacings for the sparse UCA $\mathbb{N}$ along the x, y, z-axes are $M_x d$, $M_y d$, and $M_z d$, respectively. As such, the sensors of $\mathbb{M}$ and $\mathbb{N}$ only overlap at the origin position $(0, 0, 0)$, and the total number of sensors in the cubic NCACIS is $M_x M_y M_z + N_x N_y N_z - 1$.

Assume that K uncorrelated far-field narrowband sources impinge on the cubic NCACIS from directions (θ_k, ϕ_k), where $\theta_k \in [-\pi, \pi]$ (measured counterclockwise relative to the x-axis) and $\phi_k \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ (measured bottom-to-up relative to the x-y plane) are the azimuth and elevation angles of the k-th source, $k \in \{1, 2, \cdots, K\}$. To preserve the multi-dimensional structure of cubic NCACIS signals, the total T snapshots of $\mathbb{M}$ and $\mathbb{N}$ are represented as a pair of 4-D sub-Nyquist tensors

$$\begin{aligned} \boldsymbol{\mathcal{X}}_{\mathbb{M}} &= \sum_{k=1}^{K} \boldsymbol{a}_{\mathbb{M}_x}(k) \circ \boldsymbol{a}_{\mathbb{M}_y}(k) \circ \boldsymbol{a}_{\mathbb{M}_z}(k) \circ \boldsymbol{s}_k + \boldsymbol{\mathcal{N}}_{\mathbb{M}} \in \mathbb{C}^{M_x \times M_y \times M_z \times T}, \\ \boldsymbol{\mathcal{X}}_{\mathbb{N}} &= \sum_{k=1}^{K} \boldsymbol{a}_{\mathbb{N}_x}(k) \circ \boldsymbol{a}_{\mathbb{N}_y}(k) \circ \boldsymbol{a}_{\mathbb{N}_z}(k) \circ \boldsymbol{s}_k + \boldsymbol{\mathcal{N}}_{\mathbb{N}} \in \mathbb{C}^{N_x \times N_y \times N_z \times T}, \end{aligned} \tag{1}$$

where $\boldsymbol{a}_{\mathbb{M}_x}(k) = \left[1, e^{-\jmath\pi\mu_k}, \cdots, e^{-\jmath\pi(M_x-1)\mu_k}\right]^{\mathrm{T}}$, $\boldsymbol{a}_{\mathbb{M}_y}(k) = \left[1, e^{-\jmath\pi\nu_k}, \cdots, e^{-\jmath\pi(M_y-1)\nu_k}\right]^{\mathrm{T}}$, $\boldsymbol{a}_{\mathbb{M}_z}(k) = \left[1, e^{-\jmath\pi\omega_k}, \cdots, e^{-\jmath\pi(M_z-1)\omega_k}\right]^{\mathrm{T}}$ respectively denote the steering vectors along the x, y, z-axes of $\mathbb{M}$, and $\boldsymbol{a}_{\mathbb{N}_x}(k) = \left[1, e^{-\jmath M_x\pi\mu_k}, \cdots, e^{-\jmath\pi(N_x-1)M_x\mu_k}\right]^{\mathrm{T}}$, $\boldsymbol{a}_{\mathbb{N}_y}(k) = \left[1, e^{-\jmath\pi M_y\nu_k}, \cdots, e^{-\jmath\pi(N_y-1)M_y\nu_k}\right]^{\mathrm{T}}$, $\boldsymbol{a}_{\mathbb{N}_z}(k) = \left[1, e^{-\jmath\pi M_z\omega_k}, \cdots, e^{-\jmath\pi(N_z-1)M_z\omega_k}\right]^{\mathrm{T}}$ respectively denote the steering vectors along the x, y, z-axes of $\mathbb{N}$ with $\mu_k = \cos\phi_k \cos\theta_k$, $\nu_k = \cos\phi_k \sin\theta_k$, and $\omega_k = \sin\phi_k$. Here, $\boldsymbol{s}_k = [s_k(1), s_k(2), \cdots, s_k(T)]^{\mathrm{T}}$ is the signal of the k-th source, $\boldsymbol{\mathcal{N}}_{\mathbb{M}}$, $\boldsymbol{\mathcal{N}}_{\mathbb{N}}$ are the additive Gaussian white noise tensors, i.e., $\boldsymbol{\mathcal{N}}_{\mathbb{M}(:,:,:,t)} \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \boldsymbol{\mathcal{I}})$, $\boldsymbol{\mathcal{N}}_{\mathbb{N}(:,:,:,t)} \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \boldsymbol{\mathcal{I}})$, $\forall t \in [1, T]$, σ_n^2 is the noise power, $\boldsymbol{\mathcal{I}}$ denotes an identity tensor, $\jmath = \sqrt{-1}$, $\circ$ denotes the outer product, and $(\cdot)^{\mathrm{T}}$ denotes the transpose operator.

To derive a virtual UCA, a 6-D cross-correlation tensor $\boldsymbol{\mathcal{R}} = \mathrm{E}\left\{\frac{1}{T} \boldsymbol{\mathcal{X}}_{\mathbb{M}} \underset{4}{\overset{4}{\times}} \boldsymbol{\mathcal{X}}_{\mathbb{N}}^*\right\} \in \mathbb{C}^{M_x \times M_y \times M_z \times N_x \times N_y \times N_z}$ can be calculated as

$$\boldsymbol{\mathcal{R}} = \sum_{k=1}^{K} \sigma_{s_k}^2 \boldsymbol{a}_{\mathbb{M}_x}(k) \circ \boldsymbol{a}_{\mathbb{M}_y}(k) \circ \boldsymbol{a}_{\mathbb{M}_z}(k) \circ \boldsymbol{a}_{\mathbb{N}_x}^*(k) \circ \boldsymbol{a}_{\mathbb{N}_y}^*(k) \circ \boldsymbol{a}_{\mathbb{N}_z}^*(k) + \boldsymbol{\mathcal{N}}. \tag{2}$$

Here, $\sigma_{s_k}^2 = \mathrm{E}\left\{\frac{1}{T} \boldsymbol{s}_k^{\mathrm{T}} \boldsymbol{s}_k^*\right\}$ is the power of the k-th source, $\boldsymbol{\mathcal{N}} = \mathrm{E}\left\{\frac{1}{T} \boldsymbol{\mathcal{N}}_{\mathbb{M}} \underset{4}{\overset{4}{\times}} \boldsymbol{\mathcal{N}}_{\mathbb{N}}^*\right\}$ is an all-zero cross-correlation noise tensor except the $(1,1,1,1,1,1)$-th element being σ_n^2, $\mathrm{E}\{\cdot\}$ denotes the statistical expectation, $(\cdot)^*$ denotes the conjugation operator, and $\boldsymbol{\mathcal{A}} \underset{i}{\overset{j}{\times}} \boldsymbol{\mathcal{B}}$ represents the contraction between the i-th dimension of $\boldsymbol{\mathcal{A}}$ and the j-th dimension of $\boldsymbol{\mathcal{B}}$. In practice, $\boldsymbol{\mathcal{R}}$ is approximated by the sample cross-correlation tensor $\hat{\boldsymbol{\mathcal{R}}} = \frac{1}{T} \boldsymbol{\mathcal{X}}_{\mathbb{M}} \underset{4}{\overset{4}{\times}} \boldsymbol{\mathcal{X}}_{\mathbb{N}}^*$.

III. Proposed SubTTD Method for DOA Estimation

A. Virtual Cubic Array Derivation

To derive an augmented virtual array from the sub-Nyquist tensor statistics for coarray processing, the cross-correlation tensor $\boldsymbol{\mathcal{R}}$ can be properly unfolded to combine its dimensions associated with angle information along the same directions. Specifically, we combine the dimension pairs $\{1,4\}$, $\{2,5\}$, $\{3,6\}$ of $\boldsymbol{\mathcal{R}}$, i.e., $\boldsymbol{\mathcal{R}}_{\{1,4\},\{2,5\},\{3,6\}}$, to yield a 3-D tensor

$$\begin{aligned} \boldsymbol{\mathcal{U}} = \sum_{k=1}^{K} \sigma_{s_k}^2 &\left[\boldsymbol{a}_{\mathbb{N}_x}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_x}(k)\right] \circ \left[\boldsymbol{a}_{\mathbb{N}_y}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_y}(k)\right] \\ &\circ \left[\boldsymbol{a}_{\mathbb{N}_z}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_z}(k)\right] + \boldsymbol{\mathcal{Q}} \in \mathbb{C}^{M_x N_x \times M_y N_y \times M_z N_z}, \end{aligned} \tag{3}$$

where $\boldsymbol{\mathcal{Q}} \triangleq \boldsymbol{\mathcal{N}}_{\{1,4\},\{2,5\},\{3,6\}}$ is a 3-D noise tensor, and $\otimes$ denotes the Kronecker product. Here, $\boldsymbol{a}_{\mathbb{N}_x}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_x}(k)$, $\boldsymbol{a}_{\mathbb{N}_y}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_y}(k)$, and $\boldsymbol{a}_{\mathbb{N}_z}^*(k) \otimes \boldsymbol{a}_{\mathbb{M}_z}(k)$ respectively generate consecutive difference coarrays along the x, y, z-axes. As such, a contiguous virtual UCA $\mathbb{V} = \{(x_{\mathbb{V}}, y_{\mathbb{V}}, z_{\mathbb{V}}) | x_{\mathbb{V}} \in [M_x - M_x N_x, M_x - 1]d, y_{\mathbb{V}} \in [M_y - M_y N_y, M_y - 1]d, z_{\mathbb{V}} \in [M_z - M_z N_z, M_z - 1]d\}$ can be derived.

Accordingly, rearranging the elements in $\boldsymbol{\mathcal{U}}$ to match the locations of the virtual sensors in $\mathbb{V}$ yields the equivalent second-order signals of $\mathbb{V}$, which can be modelled as a 3-D coarray tensor

$$\boldsymbol{\mathcal{V}} = \sum_{k=1}^{K} \sigma_{s_k}^2 \boldsymbol{b}_x(k) \circ \boldsymbol{b}_y(k) \circ \boldsymbol{b}_z(k) + \boldsymbol{\mathcal{Z}}. \tag{4}$$

Here, $\boldsymbol{b}_x(k) = [e^{-\jmath\pi(M_x - M_x N_x)\mu_k}, e^{-\jmath\pi(M_x - M_x N_x + 1)\mu_k}, \cdots, e^{-\jmath\pi(M_x-1)\mu_k}]^{\mathrm{T}}$, $\boldsymbol{b}_y(k) = [e^{-\jmath\pi(M_y - M_y N_y)\nu_k}, e^{-\jmath\pi(M_y - M_y N_y + 1)\nu_k}, \cdots, e^{-\jmath\pi(M_y-1)\nu_k}]^{\mathrm{T}}$, and $\boldsymbol{b}_z(k) = [e^{-\jmath\pi(M_z - M_z N_z)\omega_k}, e^{-\jmath\pi(M_z - M_z N_z + 1)\omega_k}, \cdots, e^{-\jmath\pi(M_z-1)\omega_k}]^{\mathrm{T}}$ respectively denote the steering vectors along the x, y, z-axes of $\mathbb{V}$, and $\boldsymbol{\mathcal{Z}}$ is the corresponding noise tensor.

The derived coarray tensor $\boldsymbol{\mathcal{V}}$ now enables the Nyquist-matched DOA estimation. In particular, a 6-D coarray covariance tensor of the coarray tensor $\boldsymbol{\mathcal{V}}$, i.e., $\bar{\boldsymbol{\mathcal{R}}} = \boldsymbol{\mathcal{V}}^{\mathrm{T}} \circ \boldsymbol{\mathcal{V}}^* \in \mathbb{C}^{M_x N_x \times M_y N_y \times M_z N_z \times M_x N_x \times M_y N_y \times M_z N_z}$, can be calculated as

$$\begin{aligned} \bar{\boldsymbol{\mathcal{R}}} &= \sum_{k=1}^{K} \sigma_{s_k}^4 \boldsymbol{b}_x(k) \circ \boldsymbol{b}_y(k) \circ \boldsymbol{b}_z(k) \circ \boldsymbol{b}_x^*(k) \circ \boldsymbol{b}_y^*(k) \circ \boldsymbol{b}_z^*(k) + \bar{\boldsymbol{\mathcal{Z}}} \\ &\triangleq \bar{\boldsymbol{\mathcal{D}}} \times_1 \boldsymbol{B}_x \times_2 \boldsymbol{B}_y \times_3 \boldsymbol{B}_z \times_4 \boldsymbol{B}_x^* \times_5 \boldsymbol{B}_y^* \times_6 \boldsymbol{B}_z^* + \bar{\boldsymbol{\mathcal{Z}}}, \end{aligned} \tag{5}$$

where $\bar{\boldsymbol{\mathcal{D}}} \in \mathbb{C}^{K \times K \times K \times K \times K \times K}$ is a coefficient tensor with $[\sigma_{s_1}^4, \sigma_{s_2}^4, \cdots, \sigma_{s_K}^4]$ on its diagonal, $\boldsymbol{B}_x = [\boldsymbol{b}_x(1), \boldsymbol{b}_x(2), \cdots, \boldsymbol{b}_x(K)] \in \mathbb{C}^{M_x N_x \times K}$, $\boldsymbol{B}_y = [\boldsymbol{b}_y(1), \boldsymbol{b}_y(2), \cdots, \boldsymbol{b}_y(K)] \in \mathbb{C}^{M_y N_y \times K}$, $\boldsymbol{B}_z = [\boldsymbol{b}_z(1), \boldsymbol{b}_z(2), \cdots, \boldsymbol{b}_z(K)] \in \mathbb{C}^{M_z N_z \times K}$ are the CP factor matrices, and $\times_i$ denotes the

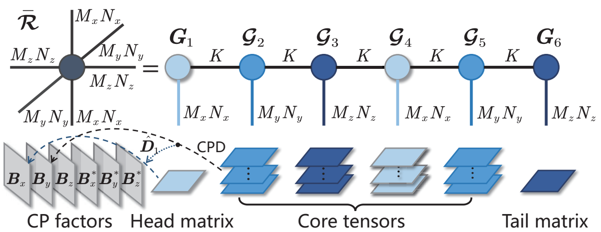

Fig. 2. Illustration of the proposed SubTTD model.

mode-i tensor-matrix product. Here, $\bar{\boldsymbol{\mathcal{Z}}} = \sum_{k=1}^{K}\sum_{k'=1,k'\neq k}^{K} \mathrm{E}\{\frac{1}{T^2}\boldsymbol{s}_k^{\mathrm{H}}(\boldsymbol{s}_k^{\mathrm{T}}\boldsymbol{s}_{k'}^{*})\boldsymbol{s}_{k'}\}\boldsymbol{b}_x(k)\circ\boldsymbol{b}_y(k)\circ\boldsymbol{b}_z(k)\circ\boldsymbol{b}_x^{*}(k')\circ\boldsymbol{b}_y^{*}(k')\circ\boldsymbol{b}_z^{*}(k')+\boldsymbol{\mathcal{Z}}^{\mathrm{T}}\circ\boldsymbol{\mathcal{Z}}^{*}$ is a residual tensor, where the cross-terms between independent signals of different sources equal to zeros, whereas the residual noise $\boldsymbol{\mathcal{Z}}^{\mathrm{T}}\circ\boldsymbol{\mathcal{Z}}^{*}$ with only one non-zero element remains.

B. Sub-Nyquist Tensor Train Formulation

Since the angle information is explicitly embedded in the CP factors of $\bar{\boldsymbol{\mathcal{R}}}$ in (5), it is necessary to retrieve these CP factors for DOA estimation. However, with the increased order of tensors, the CPD-based algorithms become increasingly difficult to converge. Thus, it will be time consuming to directly apply CPD to the 6-D coarray covariance tensor $\bar{\boldsymbol{\mathcal{R}}}$. In this regard, we propose a SubTTD model to reduce dimensionality of the coarray covariance tensor with a train of matrices and low-order tensors, based on which an efficient retrieval of the CP factors can be guaranteed.

As depicted in Fig. 2, the SubTTD represents the signal component of $\bar{\boldsymbol{\mathcal{R}}}$ as the interconnection of matrices and 3-D tensors, following a tensor train decomposition as

$$\bar{\boldsymbol{\mathcal{R}}} = \boldsymbol{G}_1 \underset{2}{\overset{1}{\times}} \boldsymbol{\mathcal{G}}_2 \underset{3}{\overset{1}{\times}} \boldsymbol{\mathcal{G}}_3 \underset{4}{\overset{1}{\times}} \boldsymbol{\mathcal{G}}_4 \underset{5}{\overset{1}{\times}} \boldsymbol{\mathcal{G}}_5 \underset{6}{\overset{1}{\times}} \boldsymbol{G}_6 + \bar{\boldsymbol{\mathcal{Z}}}, \tag{6}$$

where $\boldsymbol{G}_1 \in \mathbb{C}^{M_xN_x\times K}$ is the head matrix, $\boldsymbol{G}_6 \in \mathbb{C}^{K\times M_zN_z}$ is the tail matrix, and $\boldsymbol{\mathcal{G}}_2 \in \mathbb{C}^{K\times M_yN_y\times K}$, $\boldsymbol{\mathcal{G}}_3 \in \mathbb{C}^{K\times M_zN_z\times K}$, $\boldsymbol{\mathcal{G}}_4 \in \mathbb{C}^{K\times M_xN_x\times K}$, $\boldsymbol{\mathcal{G}}_5 \in \mathbb{C}^{K\times M_yN_y\times K}$ are core tensors representing the middle four carriages. The proposed SubTTD model in (6) can be solved by extracting dominant singular matrix subspaces for tensor train factor recovery [10]. Specifically, the head matrix $\boldsymbol{G}_1$ can be calculated from the truncated SVD $[\bar{\boldsymbol{\mathcal{R}}}]_1 = \boldsymbol{G}_1\boldsymbol{\Lambda}_1\boldsymbol{V}_1$, where $[\bar{\boldsymbol{\mathcal{R}}}]_1 \in \mathbb{C}^{M_xN_x\times M_xN_xM_y^2N_y^2M_z^2N_z^2}$ is the mode-1 unfolding of $\bar{\boldsymbol{\mathcal{R}}}$, $\boldsymbol{\Lambda}_1 \in \mathbb{C}^{K\times K}$ is the diagonal matrix with K dominant singular values, and $\boldsymbol{V}_1 \in \mathbb{C}^{K\times M_xN_xM_y^2N_y^2M_z^2N_z^2}$ is the right singular matrix. Then, the remaining product $\boldsymbol{\Lambda}_1\boldsymbol{V}_1 \in \mathbb{C}^{K\times M_xN_xM_y^2N_y^2M_z^2N_z^2}$ is reshaped into an auxiliary matrix $\boldsymbol{C}_2 \in \mathbb{C}^{M_yN_yK\times M_xN_xM_yN_yM_z^2N_z^2}$. Applying the truncated SVD to $\boldsymbol{C}_2$ yields $\boldsymbol{C}_2 = \boldsymbol{U}_2\boldsymbol{\Lambda}_2\boldsymbol{V}_2$, where $\boldsymbol{U}_2 \in \mathbb{C}^{M_yN_yK\times K}$ is the left singular matrix, $\boldsymbol{\Lambda}_2 \in \mathbb{C}^{K\times K}$ is the singular value matrix, and $\boldsymbol{V}_2 \in \mathbb{C}^{K\times M_xN_xM_yN_yM_z^2N_z^2}$ is the right singular matrix. Accordingly, the core tensor $\boldsymbol{\mathcal{G}}_2 \in \mathbb{C}^{K\times M_yN_y\times K}$ can be obtained by reshaping $\boldsymbol{U}_2$. Similarly, the SubTTD factors $\boldsymbol{\mathcal{G}}_r$ can be sequentially generated from the left singular matrices $\boldsymbol{U}_r$, $r = 3, 4, \cdots, 6$.

Different from the direct CPD on the coarray covariance tensor $\bar{\boldsymbol{\mathcal{R}}}$ which unfolds it along all dimensions to find the corresponding CP factors in an iterative loop, the proposed SubTTD solution performs the non-iterative truncated SVD and singular matrix reshaping in a sequential way to generate the SubTTD factors. Meanwhile, thanks to the train structure of the interconnected SubTTD factors, certain CP factor in (5) and SubTTD factor in (6) corresponding to the same dimension of $\bar{\boldsymbol{\mathcal{R}}}$ are embedded with angle information along the same direction. Specifically, the CP factor matrix $\boldsymbol{B}_x$ and the SubTTD head matrix $\boldsymbol{G}_1$ corresponding to the first dimension of $\bar{\boldsymbol{\mathcal{R}}}$ are embedded with angle information along the x-axis, while the CP factor matrix $\boldsymbol{B}_y$ and the SubTTD core tensor $\boldsymbol{\mathcal{G}}_2$ corresponding to the second dimension of $\bar{\boldsymbol{\mathcal{R}}}$ are embedded with angle information along the y-axis. Hence, the SubTTD factors $\boldsymbol{G}_1, \boldsymbol{\mathcal{G}}_2$ can be converted to the CP factors $\boldsymbol{B}_x, \boldsymbol{B}_y$ for DOA estimation with the least computation cost.

Algorithm 1 SubTTD-based DOA Estimation

Input: Sub-Nyquist tensors $\boldsymbol{\mathcal{X}}_{\mathbb{M}}$ and $\boldsymbol{\mathcal{X}}_{\mathbb{N}}$
Output: Estimated DOAs $\{(\theta_k, \phi_k), k = 1, 2, \cdots, K\}$
Derive the cross-correlation tensor $\boldsymbol{\mathcal{R}}$ (2), the coarray tensor $\boldsymbol{\mathcal{V}}$ (4), and the coarray covariance tensor $\bar{\boldsymbol{\mathcal{R}}}$ (5);
Compute the Truncated SVD on $[\bar{\boldsymbol{\mathcal{R}}}]_1 = \boldsymbol{G}_1\boldsymbol{\Lambda}_1\boldsymbol{V}_1$ to obtain the head matrix $\boldsymbol{G}_1$;
for $r = 2, 3, \cdots, 6$ **do**
  Reshape the product $\boldsymbol{\Lambda}_{r-1}\boldsymbol{V}_{r-1}$ into an auxiliary matrix $\boldsymbol{C}_r$;
  Compute the truncated SVD on $\boldsymbol{C}_r$ to yield $\{\boldsymbol{U}_r, \boldsymbol{\Lambda}_r, \boldsymbol{V}_r\}$;
  Derive core tensors $\boldsymbol{\mathcal{G}}_2, \cdots, \boldsymbol{\mathcal{G}}_5$ and tail matrix $\boldsymbol{G}_6$ from $\boldsymbol{U}_r$;
end for
Perform CPD on the core tensor $\boldsymbol{\mathcal{G}}_2$ to obtain $\hat{\boldsymbol{D}}_1$ and $\hat{\boldsymbol{B}}_y$ (7);
Compute $\hat{\boldsymbol{B}}_x = \boldsymbol{G}_1 \times \hat{\boldsymbol{D}}_1^{-1}$;
Retrieve $\{(\theta_k, \phi_k), k = 1, 2, \cdots, K\}$ from $\hat{\boldsymbol{B}}_x$ and $\hat{\boldsymbol{B}}_y$.

C. CP Factor Retrieval for Sub-Nyquist DOA Estimation

According to the algebraic relevance between the sequential truncated SVD-based solution for tensor train decomposition and the alternating least squares (ALS)-based solution for CPD [24], both the CP factor matrix $\boldsymbol{B}_x$ and the SubTTD head matrix $\boldsymbol{G}_1$ are calculated from the mode-1 unfolding of $\bar{\boldsymbol{\mathcal{R}}}$, but with different scaling and permutation. Hence, $\boldsymbol{B}_x$ can be obtained by multiplying $\boldsymbol{G}_1$ with a change-of-basis matrix. Meanwhile, the decomposition of the core tensor $\boldsymbol{\mathcal{G}}_2$ can not only provide such a change-of-basis matrix, but also yield the CP factor matrix $\boldsymbol{B}_y$. Therefore, we perform a CPD on $\boldsymbol{\mathcal{G}}_2$ as

$$\boldsymbol{\mathcal{G}}_2 = \boldsymbol{\mathcal{J}} \times_1 \hat{\boldsymbol{D}}_1 \times_2 \hat{\boldsymbol{B}}_y \times_3 \hat{\boldsymbol{D}}_2, \tag{7}$$

where $\boldsymbol{\mathcal{J}} \in \mathbb{C}^{K\times K\times K}$ has scaling coefficients $[\lambda_1, \lambda_2, \cdots, \lambda_K]$ on its diagonal, $\hat{\boldsymbol{D}}_1 \in \mathbb{C}^{K\times K}$, $\hat{\boldsymbol{D}}_2 \in \mathbb{C}^{K\times K}$ are the change-of-basis matrices corresponding to the first and the second dimensions of $\bar{\boldsymbol{\mathcal{R}}}$, and $\hat{\boldsymbol{B}}_y \in \mathbb{C}^{M_yN_y\times K}$ is the estimation of the CP factor matrix $\boldsymbol{B}_y$. After solving the core tensor CPD (7) through the ALS approach[2], the CP factor matrix $\boldsymbol{B}_x$ can be estimated as $\hat{\boldsymbol{B}}_x = \boldsymbol{G}_1 \times \hat{\boldsymbol{D}}_1^{-1}$.

According to the definition of the CP factor matrices in (5), μ_k and ν_k can be estimated from the phase of $\hat{\boldsymbol{B}}_x$ and $\hat{\boldsymbol{B}}_y$. Then, based on the relationship between (μ_k, ν_k) and (θ_k, ϕ_k) in Section II, the closed-form solution to the

[2] Here we implement the ALS approach as an illustrative example. Nevertheless, other optimization approaches can also be used to solve the CPD of $\boldsymbol{\mathcal{G}}_2$ (7). For more details, please see [25] and the references therein.

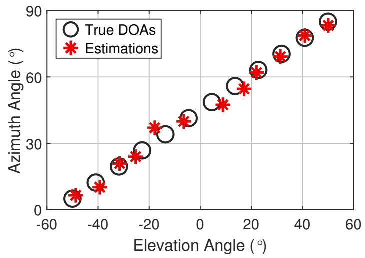

Fig. 3. Estimation performance of the proposed SubTTD-based method for 12 sources, SNR = 20 dB, $T = 800$.

azimuth and elevation angles of the k-th source can be obtained as $\hat{\theta}_k = \arctan\left(\hat{\nu}_k/\hat{\mu}_k\right)$, $\hat{\phi}_k = \arccos\left(\sqrt{\hat{\mu}_k^2 + \hat{\nu}_k^2}\right)$, $\forall k = 1, 2, \cdots, K$. The steps of the proposed SubTTD-based DOA estimation method are summarized in **Algorithm 1**.

IV. Performance Analysis

A. Computational Complexity

The proposed SubTTD-based DOA estimation method involves the sub-Nyquist tensor train formulation and the core tensor CPD procedures, whose computational complexities are $\mathcal{O}(K^2 M_x N_x M_y^2 N_y^2 M_z^2 N_z^2)$ and $\mathcal{O}(K^3 M_y N_y L_{\mathcal{G}})$, respectively. Here, $L_{\mathcal{G}}$ denotes the number of iterations for the CPD on the core tensor $\boldsymbol{\mathcal{G}}_2$ in (7). Since the CPD on a 3-D tensor normally converges within hundreds of iterations [25], the computational complexity of the proposed method is $\mathcal{O}(K^2 M_x N_x M_y^2 N_y^2 M_z^2 N_z^2)$. In contrast, the CPD-based method [6] has a computational complexity of $\mathcal{O}(K M_x^2 N_x^2 M_y^2 N_y^2 M_z^2 N_z^2 L_{\bar{\mathcal{R}}})$, where $L_{\bar{\mathcal{R}}}$ denotes the number of iterations for the direct CPD on the 6-D coarray covariance tensor $\bar{\boldsymbol{\mathcal{R}}}$ with $L_{\bar{\mathcal{R}}} \gg K$. As such, the computational complexity of the CPD-based method is $M_x N_x L_{\bar{\mathcal{R}}}/K$-times higher than that of the proposed SubTTD-based method. Thus, the proposed method enjoys a significant improvement in the computational efficiency thanks to the designed SubTTD-based coarray covariance tensor processing principle.

B. Number of Identifiable Sources

The number of identifiable sources of the proposed method is determined by the uniqueness condition of CP factor estimations from tensor train decompositions [26]. Specifically, the CP factor matrices can be uniquely retrieved from the proposed SubTTD model if and only if $\mathrm{r}(\boldsymbol{B}_x) = \mathrm{r}(\boldsymbol{B}_z) = K$, $\mathrm{r}(\boldsymbol{B}_y) \geqslant 2$, $\kappa(\boldsymbol{B}_y) \geqslant 2$, where $\mathrm{r}(\cdot)$ denotes the matrix rank, and $\kappa(\cdot)$ denotes the Kruskal's rank. This leads to $K \leqslant \min(M_x N_x, M_z N_z)$ for the proposed method, which indicates the upper bound for the number of identifiable sources.

V. Simulation Results

We consider a cubic NCACIS with $M_x{=}M_y{=}M_z = 3$ and $N_x{=}N_y{=}N_z{=}4$. The derived virtual UCA $\mathbb{V}$ has a size of $12 \times 12 \times 12$, and the number of identifiable sources is 12. First, we demonstrate the performance of the proposed method for estimating 12 sources in Fig. 3, where all the sources can be successfully located. Then, the estimation accuracy of the proposed method is compared to those of the coarray ESPRIT-based method and CPD-based method [6] in Fig. 4, where the root-mean-square error (RMSE) is used as the performance metric. For each scenario, $1,000$ Monte Carlo trials are performed. Assume that there are two distinct sources, whose azimuth and elevation angles are both randomly selected within $[15°, 60°]$ for each trial. It is clear that the proposed method achieves a better estimation accuracy in comparison with the coarray ESPRIT-based method, whose matrix-based processing fails to exploit the structural signal characteristics. Moreover, compared to the CPD-based method, the estimation accuracy of the proposed SubTTD-based method is slightly higher since it can preserve sufficient angle information while reducing the dimensionality of the coarray covariance tensor.

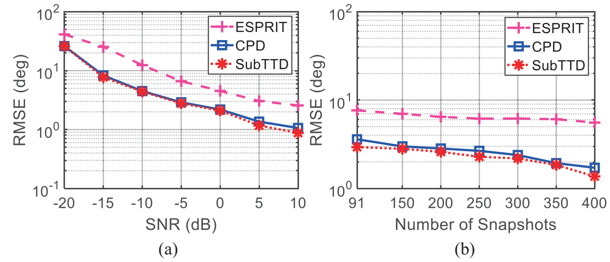

Fig. 4. Estimation accuracy comparison. (a) RMSE versus SNR, $T = 150$; (b) RMSE versus snapshots, SNR = -5 dB.

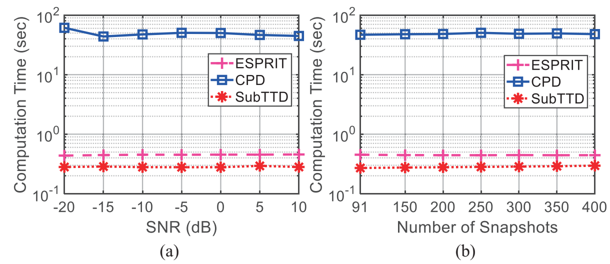

Fig. 5. Computation time comparison. (a) RMSE versus SNR, $T = 150$; (b) RMSE versus snapshots, SNR = -5 dB.

To illustrate the computational efficiency of the proposed SubTTD-based method, the computation time of the tested methods averaged from all Monte Carlo trials is depicted in Fig. 5. Obviously, the proposed method is computationally much faster than the CPD-based method, which corroborates the effectiveness of the SubTTD model to reduce computation cost for coarray tensor processing. Furthermore, benefited from the imposed dimensionality reduction strategy for the coarray covariance tensor, the proposed tensor-based processing method is even computationally faster than the coarray ESPRIT-based method with matrix-based signal processing.

VI. Conclusion

We proposed a SubTTD-based DOA estimation method for the cubic NCACIS in this letter. The formulated SubTTD model efficiently decomposes the high-order coarray covariance tensor into interconnected matrices and lower-order core tensors for retrieving the CP factors. This enables the high efficient DOA estimation with a better accuracy compared to competing matrix-based and tensor-based methods.

References

[1] P. P. Vaidyanathan and P. Pal, "Theory of sparse coprime sensing in multiple dimensions," *IEEE Trans. Signal Process.*, vol. 59, no. 8, pp. 3592–3608, Aug. 2011.

[2] Y. Gu and A. Leshem, "Robust adaptive beamforming based on interference covariance matrix reconstruction and steering vector estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3881–3885, July 2012.

[3] X. Wu, "Localization of far-field and near-field signals with mixed sparse approach: A generalized symmetric arrays perspective," *Signal Process.*, vol. 175, p. 107665, Oct. 2020.

[4] J. He, L. Li, and T. Shu, "Sparse nested arrays with spatially spread square acoustic vector sensors for high-accuracy underdetermined direction finding," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 4, pp. 2324–2336, Aug. 2021.

[5] Z. Mao, S. Liu, Y. D. Zhang, L. Han, and Y. Huang, "Joint DoA-range estimation using space-frequency virtual difference coarray," *IEEE Trans. Signal Process.*, vol. 70, pp. 2576–2592, May 2022.

[6] K. Han and A. Nehorai, "Nested vector-sensor array processing via tensor modeling," *IEEE Trans. Signal Process.*, vol. 62, no. 10, pp. 2542–2553, May 2014.

[7] C.-L. Liu and P. P. Vaidyanathan, "Tensor MUSIC in multidimensional sparse arrays," in *Proc. Asilomar Conf. Signals, Systems & Computers*, Pacific Grove, CA, Nov. 2015, pp. 1783–1787.

[8] H. Zheng, C. Zhou, Y. Gu, and Z. Shi, "Two-dimensional DOA estimation for coprime planar array: A coarray tensor-based solution," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Barcelona, Spain, May 2020, pp. 4562–4566.

[9] H. Zheng, C. Zhou, Z. Shi, M. Haardt, and J. Chen, "Coupled coarray tensor CPD for DOA estimation with coprime L-shaped array," *IEEE Signal Process. Lett.*, vol. 28, pp. 1545–1549, July 2021.

[10] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, Sept. 2011.

[11] Y. Zniyed, R. Boyer, A. L. F. de Almeida, and G. Favier, "Multidimensional harmonic retrieval based on Vandermonde tensor train," *Signal Process.*, vol. 163, pp. 75–86, Oct. 2019.

[12] F. Sedighin, A. Cichocki, T. Yokota, and Q. Shi, "Matrix and tensor completion in multiway delay embedded space using tensor train, with application to signal reconstruction," *IEEE Signal Process. Lett.*, vol. 27, pp. 810–814, Apr. 2020.

[13] X. Gong, W. Chen, J. Chen, and B. Ai, "Tensor denoising using low-rank tensor train decomposition," *IEEE Signal Process. Lett.*, vol. 27, pp. 1685–1689, Sept. 2020.

[14] J. Zhang, X. Ma, J. Qi, and S. Jin, "Designing tensor-train deep neural networks for time-varying MIMO channel estimation," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 3, pp. 759–773, Apr. 2021.

[15] K. Abed-Meraim, N. L. Trung, and R. Boyer, "Adaptive algorithms for tracking tensor-train decomposition of streaming tensors," in *Proc. European Signal Process. Conf. (EUSIPCO)*, Amsterdam, The Netherlands, Jan. 2021, pp. 995–999.

[16] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1377–1390, Mar. 2015.

[17] C. Zhou, Y. Gu, Y. D. Zhang, Z. Shi, T. Jin, and X. Wu, "Compressive sensing-based coprime array direction-of-arrival estimation," *IET Commun.*, vol. 11, no. 11, pp. 1719–1724, Aug. 2017.

[18] G. Qin, M. G. Amin, and Y. D. Zhang, "DOA estimation exploiting sparse array motions," *IEEE Trans. Signal Process.*, vol. 67, no. 11, pp. 3013–3027, June 2019.

[19] Z. Zheng, W.-Q. Wang, Y. Kong, and Y. D. Zhang, "MISC array: A new sparse array design achieving increased degrees of freedom and reduced mutual coupling effect," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1728–1741, Apr. 2019.

[20] P. Pal and P. P. Vaidyanathan, "Coprime sampling and the MUSIC algorithm," in *Proc. IEEE Digit. Signal Process. Signal Process. Educ. Meeting*, Sedona, AZ, Jan. 2011, pp. 289–294.

[21] Z. Shi, C. Zhou, Y. Gu, N. A. Goodman, and F. Qu, "Source estimation using coprime array: A sparse reconstruction perspective," *IEEE Sensors J.*, vol. 17, no. 3, pp. 755–765, Feb. 2017.

[22] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5956–5971, Nov. 2018.

[23] C. Zhou, Y. Gu, Z. Shi, and Y. D. Zhang, "Off-grid direction-of-arrival estimation using coprime array interpolation," *IEEE Signal Process. Lett.*, vol. 25, no. 11, pp. 1710–1714, Nov. 2018.

[24] Y. Zniyed, R. Boyer, A. L. F. de Almeida, and G. Favier, "High-order CPD estimation with dimensionality reduction using a tensor train model," in *Proc. European Signal Process. Conf. (EUSIPCO)*, Rome, Italy, Sept. 2018, pp. 2613–2617.

[25] P. Comon, X. Luciani, and A. L. F. de Almeida, "Tensor decompositions, alternating least squares and other tales," *J. Chemom.*, vol. 23, no. 7-8, pp. 393–405, Apr. 2009.

[26] Y. Zniyed, S. Miron, R. Boyer, and D. Brie, "Uniqueness of tensor train decomposition with linear dependencies," in *Proc. IEEE Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process. (CAMSAP)*, Guadeloupe, French West Indies, Dec. 2019, pp. 460–464.