# DOA Estimation Exploiting Sparse Array Motions

Guodong Qin, *Member, IEEE,* Moeness G. Amin, *Fellow, IEEE,* and Yimin D. Zhang, *Fellow, IEEE*

***Abstract*—This paper utilizes sparse array motion to increase the numbers of achievable both degrees of freedom (DOFs) and consecutive lags in direction-of-arrival (DOA) estimation problems. We use commonly employed environment-independent sparse array (EISA) configurations. The design of these arrays is not dependent on the sources in the field of view, but rather aims at achieving desirable difference co-arrays. They include structured coprime and nested arrays, minimum redundancy array (MRA), minimum hole array (MHA), and sparse uniform linear array (SULA). Array motion can fill the holes in the spatial autocorrelation lags associated with a fixed platform and, therefore, increases the number of sources detectable by the same number of array sensors. Quasi-stationarity of the environment is assumed where the source locations and waveforms are considered invariant over array motion of half wavelength. Closed-form expressions of the number of DOFs and consecutive spatial correlation lags for coprime and nested arrays as well as SULA, due to array translation motion, are derived. The number of DOFs and consecutive lags for the specific cases of MRA and MHA are numerically evaluated. We show the respective DOA estimation performance based on sparse reconstruction techniques.**

***Index Terms*—Sparse arrays, DOA estimation, difference co-array, coprime array, nested array, minimum redundancy array, array motion, synthetic aperture.**

## I. Introduction

SPARSE arrays provide the capability to estimate the direction-of-arrival (DOA) of more sources than the number of physical sensors [2]–[14]. They are widely utilized in many application areas, including communications, radar, sonar, satellite navigation, and radio telescope [15]–[23]. Sparse arrays can be broadly categorized as structured or non-structured arrays. Whereas the former seek the generation of large filled co-arrays, the latter are the results of optimization problems that involve environment-dependent objective functions, such as maximizing the output signal-to-noise ratio (SNR) [24], [25]. Coprime arrays and nested arrays are examples of structured arrays. It is noted that minimum redundant arrays (MRAs) and minimum hole arrays (MHAs) are in a class by themselves, as their configurations do not follow specific formula, neither do they depend on the temporal and spatial characteristics of the sources in the field of view. Rather, MRA and MHA require an exhaustive search through all possible spatial combinations of available sensors [3], [26]. In this paper, we lump MRA and MHA together with structured arrays to define a general class of environment-independent sparse arrays (EISAs).

EISAs can be mounted on a fixed or moving platform. The latter can be air-borne, vehicle-attached, or a ship-based. Moving platforms may exhibit the same sensing environment over a short observation period. Such environment includes source positions, angular directions, and signal temporal structures. In this regard, an EISA not only maintains its configuration over time, but also can operate coherently on data collected at different array positions. In so doing, the shifted array positions provide an opportunity to collectively produce difference co-arrays with a longer span and more contiguous as well as unique lags.

In this paper, we use array motion to fill the holes in the difference co-arrays associated with any fixed array position, and as such, increase the number of sources that can be estimated. Platform motion was considered in [27] and applied to coprime array with coprime integers $M$ and $N$ to provide a hole-free co-array over an extended synthetic aperture. It is shown in [27] that the coprime array must move as much as $\eta$ ($\eta$=$N$/2, $N > M$) half wavelength to produce a hole-free co-array. However, it may be unrealistic to assume a stationary signal environment over a relatively long time period, especially when considering a large array aperture. Further, fast data acquisition requirement may necessitate dealing with short observation periods. In this paper, we confine the array translation motion to only a half wavelength so that temporal and spatial source signal invariance can be reasoned.

One alternative approach to fill the missing holes and achieve increased number of lags in the difference co-array of a coprime array is through the utilization of two or multiple frequencies [26], [28]. In this case, a set of additional frequencies are employed to recover the missing lags through dilations of the co-array [26]. The number and values of the additional frequencies required for recovering the missing lags are determined by exploiting the specific structure of the co-array corresponding to the coprime configuration [26].

In this paper, considering single-frequency operation or narrowband signals with time-invariant spectra, we analyze the number of achievable degrees of freedom (DOFs) and consecutive lags as well as the respective DOA estimation performance for the case where the one-dimensional array merely moves a half wavelength along its axis. It is shown that such motion can fill most, if not all the holes, and thereby significantly increases both the contiguous and unique lags. Sparsity-based DOA estimation methods can then be applied to utilize the increased number of DOFs, which is

This work was performed while G. Qin was a Visiting Research Scholar at the Center for Advanced Communications at Villanova University. The work of Y. D. Zhang and M. G. Amin was supported in part by the National Science Foundation under Grant No. AST-1547420. Part of this work has been accepted for presentation at the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing [1].

G. Qin is with the School of Electronic Engineering, Xidian University, Xian, Shaanxi 710071, China (e-mail: gdqin@mail.xidian.edu.cn).

M. G. Amin is with the Center for Advanced Communications, College of Engineering, Villanova University, Villanova, PA 19085-1681 USA (e-mail: moeness.amin@villanova.edu).

Y. D. Zhang is with the Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 19122 USA (e-mail: ydzhang@temple.edu ).

determined by the number of unique lags [8]. Out of available compressive sensing (CS) techniques [29]–[35], which have been introduced in the literature, we choose the LASSO [31] method for source DOA estimation for moving sparse array platform.

The proposed approach can be generally applied to a sensor array operated in a quasi-stationary environment. In this paper, we consider sonar as an example, where passive synthetic aperture techniques [36]–[39] are proposed to obtain a synthetic array. The source DOAs are estimated using a moving sparse array within a relative short period.

The major contribution of this paper is two-fold: (i) We derive analytical expressions of the difference co-arrays for both nested and coprime arrays that exhibit a translation motion of a half wavelength. It is shown that the difference co-array of the combined two array positions consists of the difference co-array of the original array augmented by its half-wavelength shifted version toward and opposite to motion direction; (ii) We derive the closed-form expressions of the numbers of both DOFs and consecutive lags in the difference co-array, under array motion, for coprime and nested arrays as well as sparse uniform linear array (SULA). Specific cases for MRA and MHA are used and numerically evaluated.

The remainder of the paper is organized as follows. The basic concept of EISAs is reviewed in Section II. In Section III, the signal model and synthetic aperture processing are summarized. Section IV and V analyze the achievable numbers of both DOFs and consecutive lags after array motion for EISAs, including coprime and nested array, MRA, MHA and SULA. The maximum increments in the numbers of both DOFs and consecutive lags as a result of array motion are also derived in these two sections. Sparsity-based DOA estimation implementing LASSO is described in Section VI. In Section VII, the performance of the proposed method is evaluated through extensive simulations. Section VIII concludes the paper.

Notations: We use lower-case (upper-case) bold characters to denote vectors (matrices). In particular, $\mathbf{I}_N$ denotes the $N\times N$ identity matrix. $(\cdot)^*$ implies complex conjugation, whereas $(\cdot)^T$ and $(\cdot)^H$ respectively denote the transpose and conjugate transpose of a matrix or a vector. vec$(\cdot)$ denotes the vectorization operator that turns a matrix into a vector by stacking all columns on top of the another. $E(\cdot)$ is the statistical expectation operator and $\bigotimes$ denotes the Kronecker product. $\mathbb{S}$ denotes the sets of integers, and $\mathbb{C}$ denotes the sets of complex values.

## II. Review of EISAs

In this section, we briefly review the commonly used EISAs, namely, the structured coprime and nested arrays, and the non-structured MRA, MHA and SULA.

A coprime array consists of two subarrays with coprime number of sensors, $M$ and $N$, as illustrated in Fig. 1(a). The two subarrays share the first sensor at the zeroth position, so the total number of sensors used in the coprime array is $L = M + N - 1$. Without loss of generality, we assume $M < N$, and the sensor positions are expressed as [8]

$$\mathbb{P}_{c0} = \{Mnd, 0 \le n \le N-1\} \cup \{Nmd, 0 \le m \le M-1\}\ , \tag{1}$$

where $d$ is the unit inter-element spacing which is set to half wavelength ($\lambda/2$), with $\lambda$ denoting the wavelength.

For a nested array, we consider the two-level and the $K$-level nested arrays. The former consists of $L = N_1 + N_2$ sensors which are arranged into two uniform linear subarrays respectively of $N_1$ and $N_2$ sensors. The sensor positions are given by [3]

$$\begin{aligned}\mathbb{P}_{n2,0} =& \{nd, 0 \le n \le N_1 - 1\} \\ &\cup \{(n(N_1+1)-1)d, 1 \le n \le N_2\}\,.\end{aligned} \tag{2}$$

For an optimum $K$-level ($K = L-1$) nested array with $L$ sensors, the sensor positions are expressed as [3]

$$\mathbb{P}_{nK,0} = \{(2^n - 1)d, 0 \le n \le L-1\}\,. \tag{3}$$

The MRA and MHA are proposed to yield the largest possible co-arrays under two different constrains by searching all possible spatial position combinations of the available sensors. The sensor positions of these arrays have no close-form expressions.

We also consider the SULA with the inter-element spacing being three half-wavelengths, and the sensor positions are given by

$$\mathbb{P}_{\mathrm{u}0} = \{3nd, 0 \le n \le L-1\}\,. \tag{4}$$

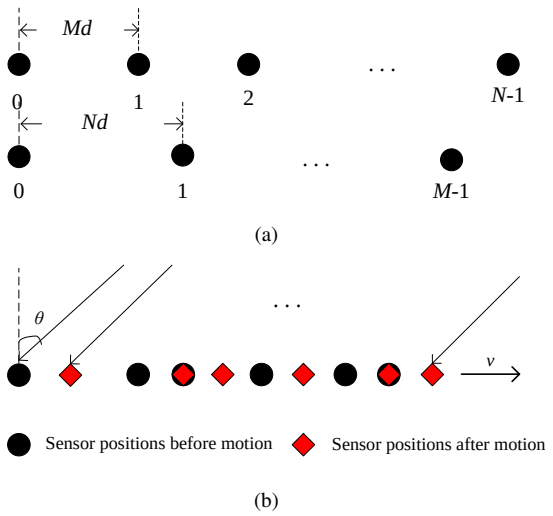



Fig. 1. DOA Estimation exploiting a moving structured sparse array. (a) The prototype coprime array; (b) The synthetic coprime array

## III. Signal Model

We consider a sparse receive array with $L$ sensors moving at a constant velocity $v$. The schematic is illustrated in Fig. 1 (b), where a coprime array is used as an example. The black circle and red rhombus represent the sensor positions of the original and shifted array, respectively.

The received signals from $Q$ far-field uncorrelated sources are described as $s_q(t)$, $t = T_s, 2T_s, \cdots, L_sT_s$, for $q =$

$1, \cdots, Q$, where $T_s$ and $L_s$, respectively, represent the sampling interval and the number of snapshots. The arrival angle of the $q$th source is denoted as $\theta_q$. Because of the assumed short translation motion of the array, the directions of the sources with respect to the sensor array can be considered fixed. The output of the receive array, at time $t$, is expressed as

$$\begin{aligned}\mathbf{x}(t) &= \sum_{q=1}^{Q} s_q(t) \exp\left(-j2\pi \frac{vt\sin(\theta_q)}{\lambda}\right)\mathbf{a}(\theta_q) + \varepsilon(t) \\ &= \mathbf{A}\mathbf{s}(t) + \varepsilon(t),\end{aligned} \tag{5}$$

where

$$\mathbf{a}(\theta_q) = \left[1, \exp(-j2\pi \frac{d_2 \sin(\theta_q)}{\lambda}), \cdots, \exp(-j2\pi \frac{d_L \sin(\theta_q)}{\lambda})\right]^T$$

is the steering vector, and $d_l$ is the position of the $l$th array sensor, $l = 1, 2, \cdots, L$. The first sensor is used as a reference, i.e., $d_1 = 0$. In addition,

$$\begin{aligned}\mathbf{s}(t) = \Bigg[&s_1(t)\exp\left(-j2\pi \frac{vt\sin(\theta_1)}{\lambda}\right), \\ &s_2(t)\exp\left(-j2\pi \frac{vt\sin(\theta_2)}{\lambda}\right), \cdots, \\ &s_Q(t)\exp\left(-j2\pi \frac{vt\sin(\theta_Q)}{\lambda}\right)\Bigg]^T\end{aligned} \tag{6}$$

is the signal vector. $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \cdots, \mathbf{a}(\theta_Q)] \in \mathbb{C}^{L\times Q}$ is the array manifold matrix, and $\varepsilon(t) \in \mathbb{C}^{L\times 1}$ is zero-mean complex additive white Gaussian noise vector with covariance matrix $\sigma_\varepsilon^2 \mathbf{I}_L$. At time $t+\tau$, the output of the receive array becomes

$$\begin{aligned}\mathbf{x}(t{+}\tau) &= \sum_{q=1}^{Q} s_q(t{+}\tau)\exp\left(-j2\pi \frac{vt\sin(\theta_q)}{\lambda}\right) \\ &\quad \cdot \exp\left(-j2\pi \frac{v\tau\sin(\theta_q)}{\lambda}\right)\mathbf{a}(\theta_q) + \varepsilon(t+\tau) \\ &= \mathbf{B}\mathbf{s}(t+\tau) + \varepsilon(t+\tau)\end{aligned} \tag{7}$$

where

$$\mathbf{B} = [\mathbf{b}(\theta_1), \mathbf{b}(\theta_1), \cdots, \mathbf{b}(\theta_Q)] \in \mathbb{C}^{L\times Q} \tag{8}$$

with

$$\begin{aligned}\mathbf{b}(\theta_q) &= \exp\left(-j2\pi \frac{v\tau\sin(\theta_q)}{\lambda}\right)\mathbf{a}(\theta_q) \\ &= \Bigg[\exp\left(-j2\pi \frac{v\tau\sin(\theta_q)}{\lambda}\right), \\ &\quad \exp\left(-j2\pi \frac{(v\tau + d_2)\sin(\theta_q)}{\lambda}\right), \cdots, \\ &\quad \exp\left(-j2\pi \frac{(v\tau + d_L)\sin(\theta_q)}{\lambda}\right)\Bigg]^T,\end{aligned} \tag{9}$$

and

$$\begin{aligned}\mathbf{s}(t+\tau) = \Bigg[&s_1(t+\tau)\exp\left(-j2\pi \frac{vt\sin(\theta_1)}{\lambda}\right), \\ &s_2(t+\tau)\exp\left(-j2\pi \frac{vt\sin(\theta_2)}{\lambda}\right), \cdots, \\ &s_Q(t+\tau)\exp\left(-j2\pi \frac{vt\sin(\theta_Q)}{\lambda}\right)\Bigg]^T.\end{aligned} \tag{10}$$

For narrowband signals with carrier frequency $f$, $s_q(t{+}\tau) = s_q(t)\exp(j2\pi f\tau)$. Accordingly, (7) can be rewritten as

$$\mathbf{x}(t{+}\tau) = \exp(j2\pi f\tau)\mathbf{B}\mathbf{s}(t) + \varepsilon(t+\tau). \tag{11}$$

By choosing $v\tau = d = \lambda/2$, the steering vector at time $t{+}\tau$ becomes

$$\begin{aligned}\mathbf{b}(\theta_q) = \Bigg[&\exp\left(-j2\pi \frac{d\sin(\theta_q)}{\lambda}\right), \\ &\exp\left(-j2\pi \frac{(d+d_2)\sin(\theta_q)}{\lambda}\right), \cdots, \\ &\exp\left(-j2\pi \frac{(d+d_L)\sin(\theta_q)}{\lambda}\right)\Bigg]^T.\end{aligned} \tag{12}$$

By compensating for the phase correction factor $\exp(j2\pi f\tau)$ using the technique described in [40], we obtain a phase synchronized received signal vector as

$$\tilde{\mathbf{x}}(t{+}\tau) = \mathbf{x}(t{+}\tau)\exp(-j2\pi f\tau) = \mathbf{B}\mathbf{s}(t) + \tilde{\varepsilon}(t+\tau), \tag{13}$$

where $\tilde{\varepsilon}(t+\tau) = \exp(-j2\pi f\tau)\varepsilon(t+\tau)$.

Combining equations (5) and (13) yields,

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \tilde{\mathbf{x}}(t{+}\tau) \end{bmatrix} = \mathbf{A}_s\mathbf{s}(t) + \begin{bmatrix} \varepsilon(t) \\ \tilde{\varepsilon}(t{+}\tau) \end{bmatrix} \in \mathbb{C}^{2L\times 1}, \tag{14}$$

where

$$\mathbf{A}_s = [\mathbf{a}_s(\theta_1), \mathbf{a}_s(\theta_2), \cdots, \mathbf{a}_s(\theta_Q)] \in \mathbb{C}^{2L\times Q}, \tag{15}$$

$$\begin{aligned}\mathbf{a}_s(\theta_q) &= [\mathbf{a}^T(\theta_q), \mathbf{b}^T(\theta_q)]^T \\ &= [1, u_2(\theta_q), \cdots, u_L(\theta_q), \\ &\quad u_d(\theta_q), u_2(\theta_q)u_d(\theta_q), \cdots, u_L(\theta_q)u_d(\theta_q)]^T,\end{aligned} \tag{16}$$

$u_l(\theta_q) = \exp\left(-j2\pi d_l \sin(\theta_q)/\lambda\right)$, and $u_d(\theta_q) = \exp\left(-j2\pi d\sin(\theta_q)/\lambda\right)$.

## IV. DOF Analysis: Coprime Array

In this section, we consider the synthetic array constructed upon moving a structured sparse array by half wavelength (one unit step) . We derive the analytical expressions of the numbers of both DOFs and consecutive lags. The former is defined as the number of non-negative unique lags in the difference co-array.

### A. Set of Cross-Lags

In the sequel, the array at its original position is referred to as the original array, whereas that in its shifted position is termed as the shifted array. The sensor positions of the shifted array are expressed as

$$\begin{aligned}\mathbb{P}_{c1} = &\{(Mn+1)d, 0 \le n \le N-1\} \\ &\cup \{(Nm+1)d, 0 \le m \le M-1\}.\end{aligned} \tag{17}$$

Combining the sensor positions of the original and the shifted arrays, we have

$$\mathbb{P}_c = \mathbb{P}_{c0} \cup \mathbb{P}_{c1}. \tag{18}$$

The original and the shifted coprime arrays are illustrated in Fig. 2 using black circles and red rhombuses, respectively. It

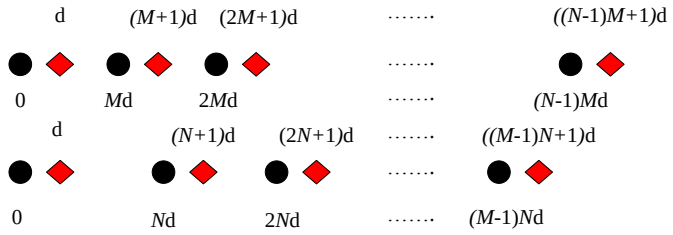



Fig. 2. The two subarrays of the synthetic coprime array.

is noted that some sensors corresponding to the original and the shifted arrays do overlap. For example, the second red rhombus in upper subarray overlaps with the second black circle in lower subarray for $N = M + 1$. This means that the number of sensors of the synthetic array becomes smaller than twice the number of sensors of the original array. Such overlapping is deemed to occur for a coprime array, because the co-array of the original coprime array contains lag-one sensor pairs [2], [8].

The set formed from the difference co-array can be divided into two subsets; one is obtained from the self-lags, whereas the other is from the cross-lags. Since the self-lag positions form a subset of the cross-lag positions [41], we only consider the cross-lags in determining the number of DOFs. The cross-lags between the two synthetic subarrays are given by the overall set:

$$\mathbb{S}_{\text{ca}} = \mathbb{S}_{12} \cup \mathbb{S}_{1'2'} \cup \mathbb{S}_{11'} \cup \mathbb{S}_{22'} \cup \mathbb{S}_{12'} \cup \mathbb{S}_{21'}. \tag{19}$$

In the above expression, subscripts 1 and 2 refer to the two subarrays in their original positions, whereas $1'$ and $2'$ refer to these subarrays in the shifted positions.

$\mathbb{S}_{12}$ is defined as the set of cross-lags between original subarray 1 and original subarray 2 which is given by [41]:

$$\mathbb{S}_{12} = \tilde{\mathbb{S}}_{12} \cup \tilde{\mathbb{S}}_{12}^{-} = \{Mk_1 - Nk_2\} \cup \{Nk_2 - Mk_1\}, \tag{20}$$

for $0 \leq k_1 \leq N-1, 0 \leq k_2 \leq M-1$, where $\tilde{\mathbb{S}}_{12} = \{Mk_1 - Nk_2\}$ and $\tilde{\mathbb{S}}_{12}^{-} = \{Nk_2 - Mk_1\}$.

Similarly,

$$\mathbb{S}_{1'2'} = \left\{Mk_1^{'} - Nk_2^{'}\right\} \cup \left\{Nk_2^{'} - Mk_1^{'}\right\}, \tag{21}$$

where $0 \leq k_1^{'} \leq N-1, 0 \leq k_2^{'} \leq M-1$. From (20) and (21), it is clear that $\mathbb{S}_{12}$ and $\mathbb{S}_{1'2'}$ have the same elements, i.e., $\mathbb{S}_{12} \cup \mathbb{S}_{1'2'} = \mathbb{S}_{12}$. On the other hand, $\mathbb{S}_{11'}$ is defined as the cross-lag set between subarray 1 and subarray $1^{'}$, whereas $\mathbb{S}_{22'}$ is defined as the cross-lag set between subarray 2 and subarray $2^{'}$. They are respectively given as,

$$\mathbb{S}_{11'} = \left\{Mk_1 - Mk_1^{'} - 1\right\} \cup \left\{Mk_1^{'} - Mk_1 + 1\right\}, \tag{22}$$

$$\mathbb{S}_{22'} = \left\{Nk_2 - Nk_2^{'} - 1\right\} \cup \left\{Nk_2^{'} - Nk_2 + 1\right\}. \tag{23}$$

Similarly,

$$\mathbb{S}_{12'} = \left\{Mk_1 - Nk_2^{'} - 1\right\} \cup \left\{Nk_2^{'} - Mk_1 + 1\right\}, \tag{24}$$

$$\mathbb{S}_{21'} = \left\{Nk_2 - Mk_1^{'} - 1\right\} \cup \left\{Mk_1^{'} - Nk_2 + 1\right\}. \tag{25}$$

*Lemma 1*: For sets $\mathbb{S}_{11'}$ , $\mathbb{S}_{22'}$ , $\mathbb{S}_{12'}$ and $\mathbb{S}_{21'}$ defined above, $\mathbb{S}_{11'} \cup \mathbb{S}_{22'} \cup \mathbb{S}_{12'} \cup 21' = \mathbb{S}_{12'} \cup \mathbb{S}_{21'}$.

*Proof.* See Appendix A. □

Because $\mathbb{S}_{12} \cup \mathbb{S}_{1'2'} = \mathbb{S}_{12}$ and based on Lemma 1, the cross-lags of the synthetic array are simplified to following set:

$$\mathbb{S}_{\text{ca}} = \mathbb{S}_{12} \cup \mathbb{S}_{12'} \cup \mathbb{S}_{21'}. \tag{26}$$

With different combinations of the subsets, it is easy to show

$$\mathbb{S}_{12'} \cup \mathbb{S}_{21'} = \tilde{\mathbb{S}}_{12'} \cup \tilde{\mathbb{S}}_{21'} \tag{27}$$

where

$$\tilde{\mathbb{S}}_{12'} = \left\{Mk_1 - Nk_2^{'} - 1\right\} \cup \left\{Nk_2 - Mk_1^{'} - 1\right\},$$

$$\tilde{\mathbb{S}}_{21'} = \left\{Nk_2^{'} - Mk_1 + 1\right\} \cup \left\{Mk_1^{'} - Nk_2 + 1\right\}.$$

Then,

$$\mathbb{S}_{\text{ca}} = \mathbb{S}_{12} \cup \tilde{\mathbb{S}}_{12'} \cup \tilde{\mathbb{S}}_{21'}. \tag{28}$$

It is noted that the new subsets $\tilde{\mathbb{S}}_{12'}$ and $\tilde{\mathbb{S}}_{21'}$ can be interpreted as the difference co-array of the original coprime array one unit step (lag) to the left and one unit step to the right. This shows that the number of unique lags increases due to motion.

We remark that if the difference co-array of the original coprime array is a filled virtual array, then there would be two additional unique lags which are created as a result of array motion, one to the left and one to the right. On the other hand, if the difference co-array of the original coprime array has holes, i.e., missing lags, the additional number of unique lags due to array motion would fill some or all of these holes. This property will be utilized to determine the number of unique lags. Because (28) does not depend on the specific subarray structures and sensor positions, the above conclusions are applicable to other sparse arrays.

### *B. Number of Unique Lags*

The number of unique lags in $\mathbb{S}_{\text{ca}}$ is the sum of the number of unique lags in $\mathbb{S}_{12}$ and the number of holes in $\mathbb{S}_{12}$ which become filled due to array motion. The number of the unique lags for CACIS is analyzed in [8]. By using a similar approach, the number of unique lags in $\mathbb{S}_{12}$ is obtained as

$$\eta_{cb} = MN + M + N - 2. \tag{29}$$

We first identify the positions of the holes in $\mathbb{S}_{12}$ for $2 < M < N$ as [8]

$$\mathbb{P}_{\text{hole}} = \{\pm(Ml_1 + Nl_2)d,\ l_1, l_2 \in \mathbb{L}\}, \tag{30}$$

where

$$\mathbb{L} = \{l_1, l_2 | 1 \leq l_1 \leq \lfloor N - 1 - \frac{N}{M} \rfloor, 1 \leq l_2 \leq \lfloor (M - \frac{2M}{N}) \rfloor, \\ |Ml_1 + Nl_2| \leq M(N-1)\}, \tag{31}$$

and $\lfloor x \rfloor$ denotes the floor function which rounds $x$ to the nearest integer towards zero. For the co-array shifted to the

$$\eta_{ci}=\begin{cases}4(M-2)-2, & N=M+1,\\ 4(N-3)-4, & N=2M-1,\\ 2[(M-1-\kappa_{2+})\alpha_{2+}+(N-4)\alpha_{2-}]-2, & N=2M+1,\\ 2[(N-1-\kappa_{1+})\alpha_{1+}+(M-1-\kappa_{1-})\alpha_{1-}]-4, & N=\beta M-1,\beta\geq 3,\\ 2[(M-1-\kappa_{2+})\alpha_{2+}+(N-1-\kappa_{2-})\alpha_{2-}]-4, & N=\beta M+1,\beta\geq 3,\\ \begin{array}{l}2[(N-1-\kappa_{1+})\alpha_{1+}+(M-1-\kappa_{1-})\alpha_{1-}+(M-1-\kappa_{2+})\alpha_{2+}\\ +(N-1-\kappa_{2-})\alpha_{2-}-1-(N-1-\kappa_{1+})(M-1-\kappa_{2+})\\ -(M-1-\kappa_{1-})(N-1-\kappa_{2-})]\\ -2\max\{\min\{M-1-\kappa_{1-},\alpha_{1+}\}\min\{N-1-\kappa_{1+},\alpha_{1-}\},\\ \qquad\min\{M-1-\kappa_{2+},\alpha_{2-}\}\min\{N-1-\kappa_{2-},\alpha_{2+}\}\},\end{array} & \text{otherwise},\end{cases}\tag{38}$$

left by one unit, the condition for the holes in set $\mathbb{S}_{12}$ to be filled is:

$$Ml_1^{'}-Nl_2^{'}-1=\pm(Ml_1+Nl_2),\tag{32}$$
$$Nl_2^{'}-Ml_1^{'}-1=\pm(Ml_1+Nl_2),\tag{33}$$

whereas the condition for the co-array shifted to the right by one unit is given by

$$Ml_1^{'}-Nl_2^{'}+1=\pm(Ml_1+Nl_2),\tag{34}$$
$$Nl_2^{'}-Ml_1^{'}+1=\pm(Ml_1+Nl_2),\tag{35}$$

where $l_1,l_2\in\mathbb{L}$, and

$$0\leq l_1^{'}\leq N-1, 0\leq l_2^{'}\leq M-1.\tag{36}$$

*Lemma 2* : For an $(L=M+N-1)$-element coprime array with unit inter-element spacing $d=\lambda/2$, the number of unique lags of the synthetic array after moving the original coprime array by a half wavelength is given by:

$$\eta_{ca}=\eta_{cb}+\eta_{ci}+2.\tag{37}$$

where $\eta_{ci}$ is the number of additional lags achieved due to array motion. It can be expressed as in (38), shown on the top of this page, where

$$\begin{aligned}\kappa_{1+}&=l_2+l_2^{'}+((N-M)(l_2+l_2^{'})+1)/M,\\ \kappa_{1-}&=l_1+l_1^{'}-((N-M)(l_1+l_1^{'})+1)/N,\\ \kappa_{2+}&=l_1+l_1^{'}+((N-M)(l_1+l_1^{'})-1)/M,\\ \kappa_{2-}&=l_2+l_2^{'}-((N-M)(l_2+l_2^{'})-1)/N.\end{aligned}\tag{39}$$

In addition, $\alpha_{1+}$ in (38) is the number of combinations that make $((N-M)(l_2+l_2^{'})+1)$ divisible by *M*, and $\alpha_{1-}$ is the number to make $(N-M)(l_1+l_1^{'})+1$ divisible by *N*. $\alpha_{2+}$ and $\alpha_{2-}$ are the numbers of combinations of $l_1,l_1^{'}$ and $l_2,l_2^{'}$ to make $(N-M)(l_1+l_1^{'})-1$ divisible by *N* and $((N-M)(l_2+l_2^{'})-1)$ divisible by *M*, respectively.

*Proof.* See Appendix B. □

### C. Number of Consecutive Lags

Note that all the holes may not be always filled due to array motion. In Fig. 3 and Fig. 4, we show two examples of an original coprime array and its corresponding synthetic array after motion. It is shown in Figs. 3(a) and 3(b) that the number of holes for both cases of $M=4$ , $N=7$ and $M=5$ , $N=6$ is the same. When considering the synthetic arrays, as shown in Fig. 4(a), the difference co-array has no holes for the case of $M=4$ , $N=7$, but Fig. 4(b) shows that one hole persists for the case of $M=5$ , $N=6$.

It is straightforward to conclude in Section IV-B that any two consecutive holes will be filled due to array motion. Accordingly, the first hole in the difference co-array of the synthetic array would appear at the first set of three consecutive holes in the difference co-array of the original synthetic array. Although the positions of maximum consecutive holes are given in [26] to determine the maximum frequency separation for an extended coprime array, yet the positions of consecutive holes are not analyzed, which is the purpose of Lemma 3.

*Lemma 3*: The position of the first positive hole in the difference co-array of the synthetic array, $p_h$, is expressed as

(a) For $N=M+1, M\geq 5$,

$$p_h=3M+N+1;$$

(b) For $M+1<N<2M-1, M\geq 7$,

$$\begin{aligned}p_h=\min[&(\kappa_{\max+}+1-\kappa_{h1+})M\\ &+(\alpha_{h1+}-\alpha_{\min+}+1)N+1,\\ &\alpha_{h1-}M+(\kappa_{\max-}+1-\kappa_{h1-})N+1];\end{aligned}$$

(c) For $N=2M-1$ and $M\geq 5$, or for $N=\beta M-1$, $\beta\geq 3$, and $M\geq 4$,

$$p_h=M+3N+1;$$

(d) For $N=\beta M+1$, $\beta\geq 2$, and $M\geq 4$,

$$p_h=(5+2(\beta-2))M+N+1;$$

(e) For $N>2M+1, N\neq\beta M\pm 1, \beta\geq 3$,

$$\begin{aligned}p_h=\min[&(\kappa_{\max}+1-\kappa_{h1+})M+\alpha_{h1+}N+1,\\ &(\alpha_{h1-}-\alpha_{\min-}+1)M+(\kappa_{\max-}+1-\kappa_{h1-})N+1]\end{aligned}$$

TABLE I
NUMBER OF UNIQUE AND CONSECUTIVE LAGS FOR COPRIME ARRAYS

| $L$ | $M$, $N$ | $\kappa_{1+}$ | $\alpha_{1+}$ | $\kappa_{1-}$ | $\alpha_{1-}$ | $\kappa_{2+}$ | $\alpha_{2+}$ | $\kappa_{2-}$ | $\alpha_{2-}$ | $\eta_{ci}$ | $\eta_{cb}$ | $\eta_{ca}$ | NULR | $\xi_{cb}$ | $\xi_{ca}$ | NCLR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 3,5 | | | | | | | | | 4 | 21 | 27 | 1.29 | 15 | 27 | 1.80 |
| 8 | 4,5 | | | | | | | | | 6 | 27 | 35 | 1.30 | 17 | 35 | 2.06 |
| 9 | 3,7 | | | | | 1 | 2 | 2 | 1 | 8 | 29 | 39 | 1.35 | 19 | 39 | 2.05 |
| 10 | 5,6 | | | | | | | | | 10 | 39 | 51 | 1.31 | 21 | 43 | 2.05 |
| | 4,7 | | | | | | | | | 12 | 37 | 51 | 1.38 | 21 | 51 | 2.43 |
| | 3,8 | 3 | 1 | 1 | 3 | | | | | 10 | 33 | 45 | 1.36 | 21 | 45 | 2.14 |
| 11 | 5,7 | 3 | 2 | 2 | 3 | 3 | 4 | 4 | 3 | 16 | 45 | 63 | 1.40 | 23 | 63 | 2.74 |
| 12 | 6,7 | | | | | | | | | 14 | 53 | 69 | 1.30 | 25 | 51 | 2.04 |
| | 5,8 | 5 | 3 | 3 | 5 | 2 | 3 | 3 | 2 | 20 | 51 | 73 | 1.43 | 25 | 73 | 2.92 |
| | 4,9 | | | | | 1 | 2 | 2 | 1 | 16 | 47 | 65 | 1.38 | 25 | 59 | 2.36 |
| | 3,10 | | | | | 1 | 3 | 3 | 1 | 14 | 41 | 57 | 1.39 | 25 | 57 | 2.28 |

(f) For other cases, no holes exist.

In the above expressions,

$$\begin{aligned}
\kappa_{h1+} &= ((N-M)(l_2^{'}+l_{n1})-1)/M + l_2^{'} + l_{n1},\\
\alpha_{h1+} &= l_2^{'} + l_{n1},\\
\kappa_{h2+} &= ((N-M)(l_2^{'}+l_{n2})-2)/M + l_2^{'} + l_{n2},\\
\alpha_{h2+} &= l_2^{'} + l_{n2},\\
\kappa_{h3+} &= ((N-M)(l_2^{'}+l_{n3})-3)/M + l_2^{'} + l_{n3},\\
\alpha_{h3+} &= l_2^{'} + l_{n3},\\
\kappa_{\max+} &= \max(\kappa_{h1+}, \kappa_{h2+}, \kappa_{h3+}),\\
\alpha_{\min+} &= \min(\alpha_{h1+}, \alpha_{h2+}, \alpha_{h3+}).
\end{aligned}$$

Similarly,

$$\begin{aligned}
\kappa_{h1-} &= l_1^{'} + l_{m1} - ((N-M)(l_1^{'}+l_{m1})+1)/N\\
\alpha_{h1-} &= l_1^{'} + l_{m1},\\
\kappa_{h2-} &= l_1^{'} + l_{m2} - ((N-M)(l_1^{'}+l_{m2})+2)/N\\
\alpha_{h2-} &= l_1^{'} + l_{m2},\\
\kappa_{h3-} &= l_1^{'} + l_{m3} - ((N-M)(l_1^{'}+l_{m3})+3)/N,\\
\alpha_{h3-} &= l_1^{'} + l_{m3},\\
\kappa_{\max-} &= \max(\kappa_{h1-}, \kappa_{h2-}, \kappa_{h3-}),\\
\alpha_{\min-} &= \min(\alpha_{h1-}, \alpha_{h2-}, \alpha_{h3-}).
\end{aligned}$$

*Proof.* See Appendix C. □

By definitions, the number of consecutive lags is $\xi_{ca} = 2p_h - 1$, which becomes $\xi_{ca} = 2M(N-1)+3$ if there are no holes. This is because the lags of the difference co-array, corresponding to the original array, are distributed between $-M(N-1)$ and $M(N-1)$. On the other hand, $\xi_{cb} = 2(N+M)-1$.

*Example 1*: Table I deals with co-prime arrays and shows examples of the parameters computed from Lemma 2 and Lemma 3 for different pairs of $M$ and $N$. NULR denotes the ratio between the numbers of unique lags with and without array motion, i.e., $\eta_{ca}/\eta_{cb}$. NCLR denotes the ratio between the numbers of consecutive lags with and without array motion. The difference co-arrays corresponding to the original and synthetic coprime arrays are respectively presented in Fig. 3 and Fig. 4.

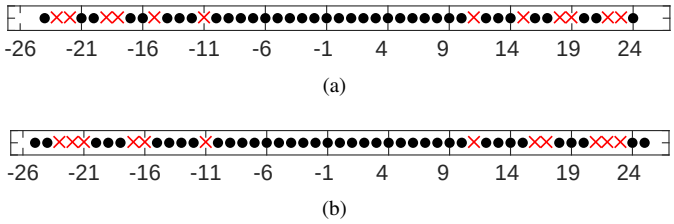



Fig. 3. Difference co-array of original coprime array. (a) $M$=4, $N$=7; (b) $M$=5, $N$=6.(●:lags; ×:holes)

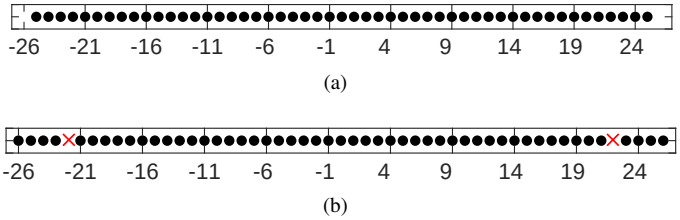



Fig. 4. Difference co-array of synthetic coprime array. (a) $M$=4, $N$=7; (b) $M$=5, $N$=6.

From Table I, Fig. 3 and Fig. 4, we observe that the numbers of both unique and consecutive lags increase substantially due to array motion. This is largely because that all (see Fig. 4(a)) or most (see Fig. 4(b)) holes become filled. The maximum number of additional unique and consecutive lags are obtained when $2M-1 \geq N > M+1$ for a given number of sensors.

## V. DOF Analysis: Other Sparse Arrays

### A. Nested Array

Similar to the coprime array, the difference co-array of the nested array after moving one unit step consists of the difference co-array before motion and the results of moving one unit step to the left and 1 step to the right from the original difference co-array. Because the difference co-array of two-level nested array is fully filled, the number of unique lags after the motion is expressed as

$$\eta_{na,2} = \eta_{nb,2} + 2, \tag{40}$$

where the expression of $\eta_{nb,2}$ is given in Table II [3].

For an optimum *K*-level nested array ($K = L-1$), the number of unique lags before the array motion is given by $\eta_{nb,K} = L(L-1)+1$ [3]. The number of unique lags after the array motion is expressed as

$$\eta_{na,K} = 3L^2 - 13L + 21, \quad L \geq 4. \tag{41}$$

The derivation of (41) is provided below.

The lags in the difference co-arrays of the original and the synthetic *K*-level nested arrays are respectively given by sets $\mathbb{S}_{nb,K}$ and $\mathbb{S}_{na,K}$, expressed as

$$\begin{aligned} \mathbb{S}_{nb,K} &= 2^{n_1} - 2^{n_2}, \\ \mathbb{S}_{na,K} &= \{2^{n_1} - 2^{n_2} + 1\} \cup \{2^{n_1} - 2^{n_2} - 1\}, \end{aligned} \tag{42}$$

where $n_1, n_2 \in [0, L-1]$.

From the above equation, we find that $\mathbb{S}_{nb,K}$ and $\mathbb{S}_{na,K}$ form subsets of $\mathbb{S}_{nb,K+1}$ and $\mathbb{S}_{na,K+1}$, respectively. Therefore, $\eta_{na,K+1}$ can be determined based on $\eta_{na,K}$ and the number of additional lags offered through the array motion. We denote $\eta_{nb,K}$ and $\eta_{na,K}$ as the number of the entire original co-array and that of the original difference co-array, respectively. For the convenience of description, we only consider the number of additional lags rendered in the non-negative difference co-array.

For the original difference co-array, $\eta_{nb,K}$ is equal to 13 for $L = 4$. There exists only a single hole in the non-negative difference co-array located at the 5th lag position, together with a symmetric hole in the negative difference co-array. As a result, the number of unique lags after array motion becomes $\eta_{na,K} = 17$ as a result of filling these holes and extending the co-array aperture by one unit to both the left and the right. When the level is increased by 1, we obtain $L-1$ additional lags located at

$$\mathbb{S}_{ad} = \{2^{L-1}-2^{L-2},\ 2^{L-1}-2^{L-3},\ \cdots,\ 2^{L-1}-2,\ 2^{L-1}-1\} \tag{43}$$

in the non-negative difference co-array, while the other positions between $2^{L-1} - 2^{L-2}$ and $2^{L-1} - 1$ are holes. From $\mathbb{S}_{ad}$, we can obtain the difference set of the adjacent lags as $\mathbb{S}_d = \{2^{L-3}, 2^{L-4}, \cdots, 4, 2, 1\}$. It implies that, except the single hole between 2 and 4, other holes form $L-4$ groups with 3 or more consecutive holes in each group.

The array motion fills in the single hole, and reduces the number of missing lags in each consecutive hole group by 2. As a result, for each addtional sensor, $L-1$ lags are added in the original co-array, whereas $1 + 2(L-4)$ lags are added by filling the holes after the array motion. In addition, one additional lag at $2^{L-1}$ is obtained by moving the co-array to the right. Note that this lag always overlaps with the the smallest lag in $S_{ad}$ in the $(L+1)$-sensor array, i.e., $2^L - 2^{L-1} = 2^{L-1}$, thus reducing the lag increment by 1 when an additional level is added.

As the result of the above discussion, each additional sensor yields a total increment of $3L-8$ lags in the resulting non-negative synthetic co-array. Considering $\eta_{na,K} = 17$ for $L = 4$, we can obtain $\eta_{na,K} = 3L^2 - 13L + 21$ for $L \geq 4$ as shown in (41).

TABLE II
THEORETICAL EXPRESSIONS OF THE NUMBER OF UNIQUE LAGS OF TWO-LEVEL NESTED ARRAY

| $L$ | Optimal $N_1$,$N_2$ | $\eta_{nb,2}$ |
|---|---|---|
| Even | $N_1 = N_2 = L/2$ | $(L^2-2)/2 + L$ |
| Odd | $N_1 = (L-1)/2$, $N_2 = (L+1)/2$ | $(L^2-1)/2 + L$ |

TABLE III
NUMBER OF UNIQUE LAGS OF A MOVING MRA

| $L$ | $\eta_{ri}$ | $\eta_{rb}$ | $\eta_{ra}$ | NULR | $\xi_{rb}$ | $\xi_{ra}$ | NCLR |
|---|---|---|---|---|---|---|---|
| 3 | 2 | 7 | 9 | 1.29 | 7 | 9 | 1.29 |
| 5 | 2 | 19 | 21 | 1.11 | 19 | 21 | 1.11 |
| 6 | 2 | 27 | 29 | 1.07 | 27 | 29 | 1.07 |
| 8 | 2 | 47 | 49 | 1.04 | 47 | 49 | 1.04 |
| 9 | 2 | 59 | 61 | 1.03 | 59 | 61 | 1.03 |

TABLE IV
NUMBER OF UNIQUE LAGS OF A MOVING MHA

| $L$ | $\eta_{hi}$ | $\eta_{hb}$ | $\eta_{ha}$ | NULR | $\xi_{hb}$ | $\xi_{ha}$ | NCLR |
|---|---|---|---|---|---|---|---|
| 3 | 2 | 7 | 9 | 1.29 | 7 | 9 | 1.29 |
| 5 | 4 | 21 | 25 | 1.191 | 19 | 25 | 1.32 |
| 6 | 6 | 31 | 37 | 1.193 | 27 | 37 | 1.37 |
| 8 | 14 | 57 | 71 | 1.246 | 31 | 71 | 2.29 |
| 9 | 18 | 73 | 91 | 1.247 | 35 | 91 | 2.60 |

We now can determine the position of the first hole in the difference co-array of the *K*-level nested array. When $L = 4$, the first hole appears at the 5th position of the original *K*-level nested array. When $L = 5$, from the expression of $\mathbb{S}_{ad}$ in (43), the position of the middle one of the first three consecutive holes is $2^{L-1} - 6 = 10$. Because $\mathbb{S}_{nb,K}$ and $\mathbb{S}_{na,K}$ respecxtively form subsets of $\mathbb{S}_{nb,K+1}$ and $\mathbb{S}_{na,K+1}$, this first hole position at lag 10 is unchanged as the number of sensors increases, as can be seen in Fig. 5(b) and Fig. 6(b).

### *B. MRA and MHA*

Because MRA and MHA do not have closed-form expressions for their array geometry and the number of achievable DOFs, we analyze their DOFs using examples, illustrated in Table III and Table IV. The same sensor positions as in [42] are used. Because there may be multiple array configurations for MHA and MRA for a given number of sensors, the numbers of consecutive lags ($\xi_{rb}$ and $\xi_{hb}$) are set to the maximum value.

### *C. SULA*

For the uniform sparse array with the inter-element spacing being three half wavelength, the sensor positions of after array motion are given by

$$\mathbb{P}_{u1} = \{3nd + 1,\ 0 \leq n \leq L-1\}. \tag{44}$$

Because of the specific inter-element spacing, the number of unique lags in the difference array equals to $2(L-1)+1$, i.e., $\eta_{sb} = 2L-1$, and the number of holes in the difference array is $4(L-1)$. The number of the lags in the difference co-array of the synthetic array is expressed as

$$\eta_{sa} = 3(2L-1). \tag{45}$$

The difference co-array after the array motion is fully filled because the number of holes between each pair of adjacent sensors is two. It is noted that, unlike other arrays, $\eta_{sa}/\eta_{sb} = 3$ is constant for the SULA.

*Example 2*: Difference co-arrays of the original and synthetic arrays are respectively presented in Fig. 5 and Fig. 6 for the nested array, MRA, MHA and SULA. From the figure, we observe again that the numbers of both unique and consecutive lags increase after array motion. Especially, for the SULA, there are no consecutive lags before array motion, whereas the difference co-array is fully filled after the array motion.

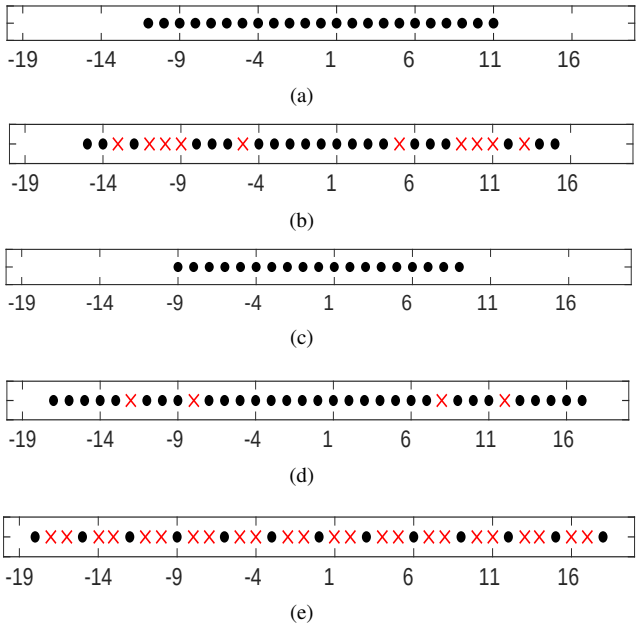



Fig. 5. Difference co-arrays of original sparse arrays. (a) Two-level nested array with $L = 6$; (b) Four-level nested array with $L = 5$; (c) MRA with $L = 5$; (d) MHA with $L = 6$; (e) SULA with $L = 7$.

### D. Remarks

So far, we have analyzed the achievable DOFs and the consecutive lags for different structured sparse arrays with array motion. The theoretical expressions are summarized in Table V.

The following points are worth noticing:

1) The difference co-arrays of the synthetic two-level nested array, MRA and SULA are all fully filled. The first two original sparse arrays have fully-filled difference co-arrays, while all the lags of the original SULA are disjoint.
2) The SULA achieves the highest NULR of 3, whereas the two-level nested array and MRA perform worst on the NULR and NCLR because only exactly 2 lags are added with the array motion, regardless of the number of sensors. The NULR and NCLR of the coprime array vary and achieve their maximum values when $2M-1 \geq N > M+1$ (see Table I). The NULR of the $K$-level nested array increases as the number of sensors increases because the term $10L - 18/L^2 - L + 1$ monotonically decreases with $L$. The same situation happens on MHA.

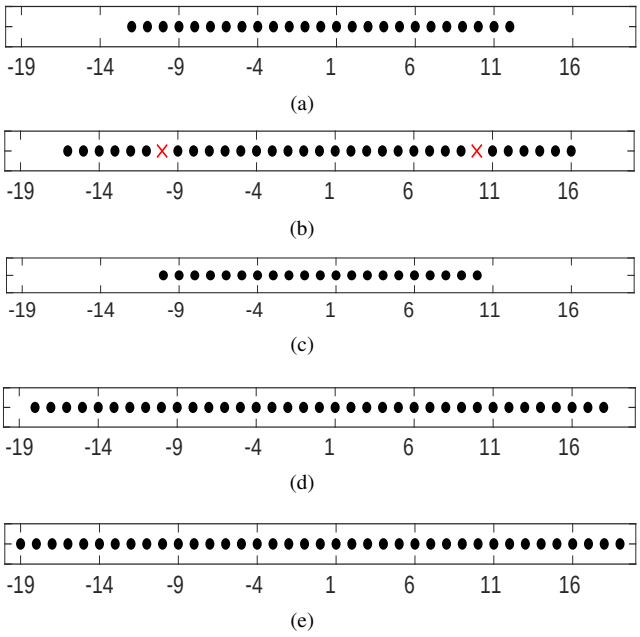



Fig. 6. Difference co-array of synthetic sparse arrays. (a) Two-level nested array with $L = 6$; (b) Four-level nested array with $L = 5$; (c) MRA with $L = 5$; (d) MHA with $L = 6$; (e) SULA with $L = 7$.

3) For a given number of sensors, the $K$-level nested array achieves the highest number of unique lags by exploiting array motion because more holes in the original difference co-array are filled by the array motion. However, it has the smallest number of consecutive lags, which is equal to 19. MHA has the highest number of consecutive lags. The $K$-level nested array works better than SULA for a given number of sensors.

## VI. DOA Estimation

In this section, we briefly introduce the DOA estimation based on compressive sensing [7] and co-array MUSIC [3], [6], which utilize all and consecutive lags, respectively.

From (14), the covariance matrix of $\mathbf{y}(t)$ is expressed as

$$\mathbf{R_y} = \mathbf{A}_s \mathbf{R_s} \mathbf{A}_s^H + \sigma_\varepsilon \mathbf{I}_{\tilde{\mathbf{L}}}, \tag{46}$$

where $\mathbf{R_s} = E[\mathbf{s}(t)\mathbf{s}^H(t)] = \text{diag}[\sigma_1, \sigma_2, \cdots, \sigma_Q]$ is the source covariance matrix with $\sigma_q$ real and positive, and $\sigma_\varepsilon$ denotes the noise variance. In addition, $\tilde{L}$ is the number of sensors in the synthetic array. Note that $\tilde{L} \leq 2L$ because some sensor positions before and after motion may overlap. Vectorizing $\mathbf{R_y}$ yields the following $\tilde{L}^2 \times 1$ vector

$$\tilde{\mathbf{z}} = \text{vec}(\mathbf{R_y}) = \tilde{\mathbf{A}}\tilde{\mathbf{b}} + \sigma_\varepsilon \mathbf{i}, \tag{47}$$

where

$$\begin{aligned} \tilde{\mathbf{A}} &= [\tilde{\mathbf{a}}(\theta_1), \tilde{\mathbf{a}}(\theta_2), \cdots, \tilde{\mathbf{a}}(\theta_Q)] \in \mathbb{C}^{\tilde{L}^2 \times Q}, \\ \tilde{\mathbf{a}}(\theta_q) &= \mathbf{a_s}^*(\theta_q) \otimes \mathbf{a_s}(\theta_q) \in \mathbb{C}^{\tilde{L}^2 \times 1}, \\ \tilde{\mathbf{b}} &= [\sigma_1, \sigma_2, \cdots, \sigma_Q]^T, \\ \mathbf{i} &= \text{vec}(\mathbf{I}_{\tilde{L}}) = [10\cdots0, 01\cdots0, \cdots, 00\cdots1]^T \in \mathbb{R}^{\tilde{L}^2 \times 1}. \end{aligned} \tag{48}$$

TABLE V
THEORETICAL EXPRESSIONS OF ACHIEVABLE DOFS AND CONSECUTIVE LAGS FOR DIFFERENT STRUCTURED SPARSE ARRAYS WITH ARRAY MOTION

| | **Coprime array** | **Two-level nested array** | $K$**-level nested array** | **MRA** | **MHA** | **SULA** |
|---|---|---|---|---|---|---|
| **Number of unique lags of original array** | $MN+M+N-2$ | $\frac{L^2-2}{2}+L, L$ is even<br>$\frac{L^2-1}{2}+L, L$ is odd | $L(L-1)+1$ | see Table III | see Table IV | $2L-1$ |
| **Number of consecutive lags of original array** | $2(M+N)-1$ | $\frac{L^2-2}{2}+L, L$ is even<br>$\frac{L^2-1}{2}+L, L$ is odd | 9 | see Table III | see Table IV | 0 |
| **Number of unique lags of synthetic array** | $MN+M+N+\eta_{ci}$ | $\frac{L^2+2}{2}+L, L$ is even<br>$\frac{L^2+3}{2}+L, L$ is odd | $3L^2-13L+21$ | see Table III | see Table IV | $3(2L-1)$ |
| **Number of consecutive lags of synthetic array** | $2p_h-1, p_h\neq 0$<br>$2M(N-1)+3, p_h=0$ | $\frac{L^2+2}{2}+L, L$ is even<br>$\frac{L^2+3}{2}+L, L$ is odd | 19 | see Table III | see Table IV | $3(2L-1)$ |
| **NULR** | $1+\frac{\eta_{ci}+2}{MN+M+N-2}$ | $1+\frac{4}{L^2+2L-2}, L$ is even<br>$1+\frac{4}{L^2+2L-1}, L$ is odd | $3-\frac{10L-18}{L^2-L+1}$ | see Table III | see Table IV | 3 |
| **NCLR** | $\frac{2p_h-1}{2(M+N)-1}, p_h\neq 0$<br>$\frac{2M(N-1)+3}{2(M+N)-1}, p_h=0$ | $1+\frac{4}{L^2+2L-2}, L$ is even<br>$1+\frac{4}{L^2+2L-1}, L$ is odd | 2.11 | see Table III | see Table IV | N/A |

To utilize all the lags in the difference co-array, (47) can be replaced by the sparse signal representation

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{B}}\mathbf{r} + \sigma_\varepsilon \mathbf{i} \tag{49}$$

where $\tilde{\mathbf{B}} = [\tilde{\mathbf{a}}^T(\theta_1), \tilde{\mathbf{a}}^T(\theta_2), \cdots, \tilde{\mathbf{a}}^T(\theta_{\tilde{Q}})]^T$, and $\mathbf{r}$ is a $\tilde{Q} \times 1$ vector to be determined. The positions and values of the non-zero entries in $\mathbf{r}$ represent the estimated DOAs and the corresponding signal power. $\tilde{Q} \gg Q$ denotes the size of the search grid in spatial angles.

The estimation is obtained by solving the following LASSO problem [31]

$$\hat{\mathbf{r}} = \arg\min_{\mathbf{r}} \left[ \frac{1}{2} \left\| \tilde{\mathbf{Y}} - \mathbf{B}\mathbf{r} \right\|_2^2 + \lambda_t \|\mathbf{r}\|_1 \right], \tag{50}$$

where $\lambda_t$ is a regularization parameter, and $\|\cdot\|_2$ and $\|\cdot\|_1$ denote the $l_2$-norm and $l_1$-norm, respectively.

Using the consecutive lags in the difference co-array, the co-array MUSIC can be applied to obtain DOA estimates. This technique proceeds by constructing the full rank matrix $\mathbf{R}_{ss}$, expressed as [6],

$$\mathbf{R}_{ss} = \frac{1}{L_{ss}} \sum_{p=1}^{L_{ss}} \mathbf{J}_p \mathbf{y}_{\text{diff}}^{\text{ULA}} \{\mathbf{y}_{\text{diff}}^{\text{ULA}}\}^H \mathbf{J}_p^H. \tag{51}$$

where $L_{ss} = (L_{\text{ULA}}+1)/2$ with $L_{\text{ULA}}$ denoting the number of contiguous lags in the difference co-array, and $\mathbf{y}_{\text{diff}}^{\text{ULA}}$ is the respective coarray data vector. In addition, $\mathbf{J}_p$ is a selection matrix defined as [6]

$$\begin{aligned} \mathbf{J}_p = &\left[\mathbf{0}_{L_{ss}\times(L_{ss}-1-p)} \quad \mathbf{I}_{L_{ss}\times L_{ss}} \quad \mathbf{0}_{L_{ss}\times p}\right] \\ &\in \{0,1\}^{L_{ss}\times(2L_{ss}-1)}. \end{aligned} \tag{52}$$

where $\mathbf{0}_{L_{ss}\times p}$ is a $L_{ss} \times p$ zero matrix.

## VII. NUMERICAL RESULTS

In this section, we present DOA estimation results through Monte Carlo simulations. The RMSE of the estimated DOA of the sources, expressed as

$$\text{RMSE} = \sqrt{\frac{1}{QN_m} \sum_{p=1}^{N_m} \sum_{q=1}^{Q} \left(\hat{\theta}_q(p) - \theta_q\right)^2} \tag{53}$$

is used for performance comparison, where $\hat{\theta}_q(p)$ is the estimate of $\theta_q$ for the $p$th Monte Carlo trial, $N_m$ is the number of the Monte Carlo trials, and $N_m = 100$ in all simulations.

### A. Achievable Number of DOFs

In this simulation, we analyze the achievable number of DOFs for the coprime array, $K$-level nested array, MHA and SULA. The two-level nested array and MRA are not included because their difference co-arrays are fully filled, and the additional number of unique lags is only 2. In all the simulations, the input SNR is set to 10 dB, and 1,000 snapshots are used. Because each array yields a different number of DOFs, different values of $Q$ are utilized. In the first simulation, the LASSO algorithm [31] is used to perform sparsity-based DOA estimation, and all lags are exploited. $Q = 21$ sources are used for the coprime array with $M = 5$ and $N = 6$, whereas the value of $Q$ is taken as 13, 18 and 17 respectively for the four-level nested array, 6-element MHA and 8-element SULA. For the MHA, the sensor positions are [0, 1, 4, 10, 15, 17]. The spatial frequencies of the $Q$ sources, denoted as $\tilde{\theta} = d\sin(\theta)/\lambda$, are uniformly distributed between $-0.49$ and $0.49$, which correspond to $-78.5^\circ$ and $78.5^\circ$, respectively.

The simulations results are shown in Fig. 7 and Fig. 8. It is clear from Fig. 7 that the original sparse array cannot identify the DOAs of all $Q$ sources because of insufficient DOFs. Grating lobes appear in Fig. 7(d) because the SULA has large inter-element spacing. On the other hand, as shown in Fig. 8, the proposed method correctly identify the DOAs of all $Q$ sources as a result of the additional DOFs achieved through array motion.

In the second simulation example, we use the Co-array MUSIC and only the consecutive lags are used. The values of $Q$ are set to 18, 7 and 15, respectively, for the coprime array, the four-level nested array and the 6-element MHA. The other parameters remain the same as in the first simulation. The simulations results are shown in Fig. 9. After motion, in Figs. 9(b), 9(d) and 9(f), the number of consecutive lags increases, thus enabling the synthetic array to identify the DOAs of all $Q$ sources. On the other hand, in Figs. 9(a), 9(c) and 9(e), the

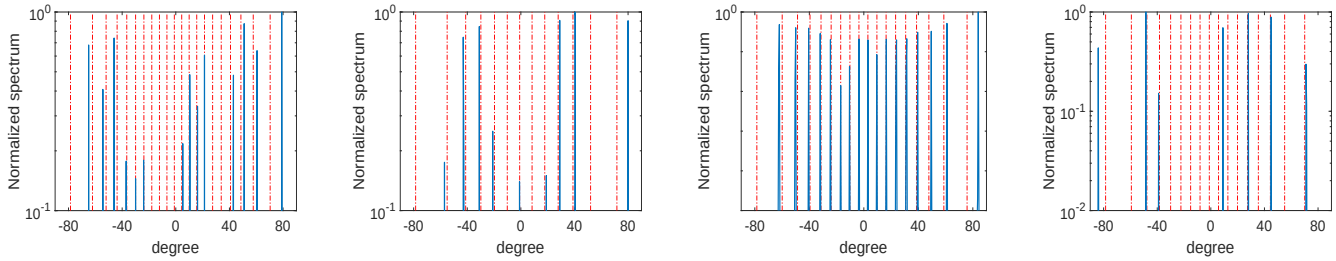



Fig. 7. Estimated DOAs for original sparse arrays using LASSO. (a) Coprime array with $M = 5$ and $N = 6$ for 21 sources, $\eta_{cb} = 39$; (b) Four-level nested array with $L = 5$ for 13 sources, $\eta_{nb,K} = 21$; (c) MHA with $L = 6$ for 18 sources, $\eta_{hb} = 31$; (d) SULA with $L = 8$ for 17 sources, $\eta_{sb} = 17$.

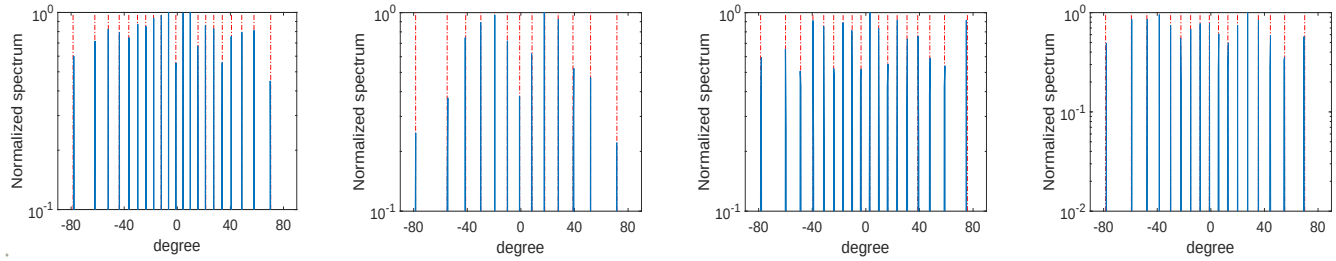



Fig. 8. Estimated DOAs for synthetic sparse arrays using LASSO. (a) Coprime array with $M = 5$ and $N = 6$ for 21 sources, $\eta_{ca} = 51$; (b) Four-level nested array with $L = 5$ for 13 sources, $\eta_{na,K} = 31$; (c) MHA with $L = 6$ for 18 sources, $\eta_{ha} = 37$; (d) SULA with $L = 8$ for 17 sources, $\eta_{sa} = 45$.

$Q$ sources are not estimated correctly. This result is similar to the results obtained from LASSO, and the reason lies in the increased number of consecutive and unique lags through array motion.

### *B. Estimation performance analysis*

We compare the RMSE performance for different sparse arrays based on LASSO and co-array MUSIC. 2,000 snapshots of data are used, and the results are averaged with 100 Monte Carlo trails. The input SNR varies between $-10$ dB and 20 dB, and the grid is set to $0.02^\circ$. The numbers of sensors and sources are, respectively, set to 7 and 6 for all sparse arrays. For the coprime array, $M = 3$ and $N = 5$ are assumed, the first sensor of the two subarrys is overlapping. For LASSO, the RMSE performance versus the input SNR for the original and the synthetic arrays is shown in Fig. 10 using dash dot lines and solid lines, respectively. The superiority of the synthetic array over the original array is evident. The six-level nested array offers the highest improvement with most additional lags through array motion, whereas the MRA yields the smallest increment of 2 additional lags in the co-array after array motion. Regarding the RMSE performance, without the array motion, the six-level nested array provides the best estimation performance, while the SULA performs the worst. On the other hand, after the array motion, the six-level nested array provides the best estimation performance, while the coprime array performs the worst. Although the original MHA and the six-level nested array have the same number of unique lags, the latter has a larger array aperture and thus yields better estimation performance. The synthetic SULA, the original and the synthetic MRAs have similar number of unique lags as well as array aperture and thus lead to close RMSE performance. Fig. 11 shows the RMSE performance as the number of snapshots changes for the original and the synthetic arrays. In this simulation, the input SNR is set to 5 dB, and the number of snapshot varies from 200 to 6,000. The other parameters are same as those used in Fig. 10. Due to the difference of the number of unique lags and array aperture, before the array motion, the the six-level nested array performs the best, whereas SULA performs worst. After the array motion, the nested array performs the best and the coprime array does the worst. The other sparse arrays perform similarly to Fig. 10.

The co-array MUSIC uses the same parameter values as in LASSO. Here, the 2-level nested array is adopted to replace the $K$-level nested array since the latter only identifies 5 sources before array motion. The simulation RMSE results with respect to the input SNR and the number of snapshots are respectively shown in Fig. 12 and Fig. 13. It is clear that the synthetic array is superior to the original array due to the increased number of the consecutive lags. The MHA obtains the highest performance improvement with the highest additional consecutive lags after array motion, whereas the MRA and 2-level nested array offer the lowest increment of 2 additional lags. It is noted that before the array motion, the MRA performs the best, while the MHA and the coprime arrays perform the worst. The synthetic and original arrays for the 2-level nested array and the MRA have similar estimation performance because the numbers of the consecutive lags corresponding to the synthetic and original array are very close.

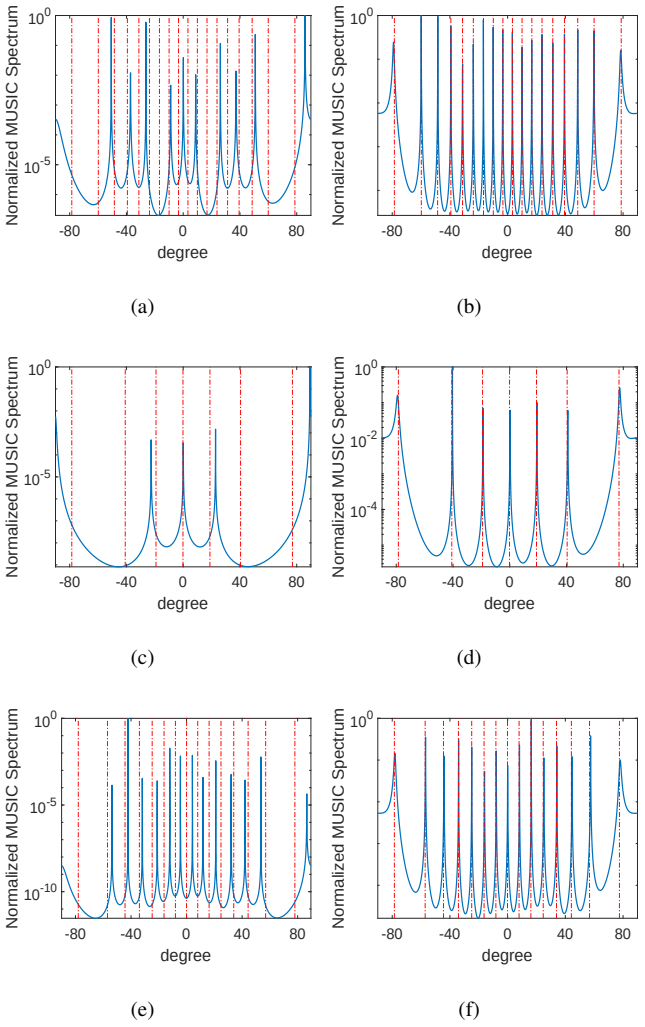



Fig. 9. Estimated DOAs using co-array MUSIC. (a) The original coprime array with $M = 5$ and $N = 6$ for 18 sources, $\xi_{cb} = 21$; (b) The synthetic coprime array with $M = 5$ and $N = 6$ for 18 sources, $\xi_{ca} = 43$; (c) The Original four-level nested array with $L = 5$ for 7 sources, $\xi_{nb} = 9$; (d) The synthetic four-level nested array with $L = 5$ for 7 sources, $\xi_{na} = 19$; (e) The original MHA with $L = 6$ for 15 sources, $\xi_{hb} = 27$; (f) The synthetic MHA with $L = 6$ for 15 sources, $\xi_{ha} = 37$. ($\xi_{nb}$ and $\xi_{na}$ are the number of consecutive lags for the original and synthetic four-level nested array.)

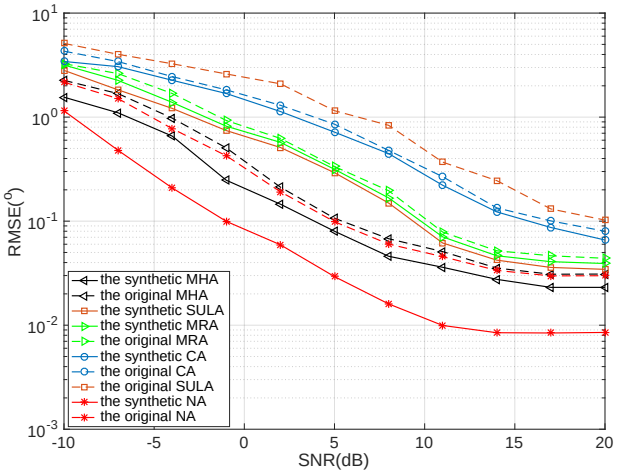



Fig. 10. RMSE vs. SNR for different sparse arrays using LASSO. (CA and NA denote coprime array and $K$-level nested array, respectively.)

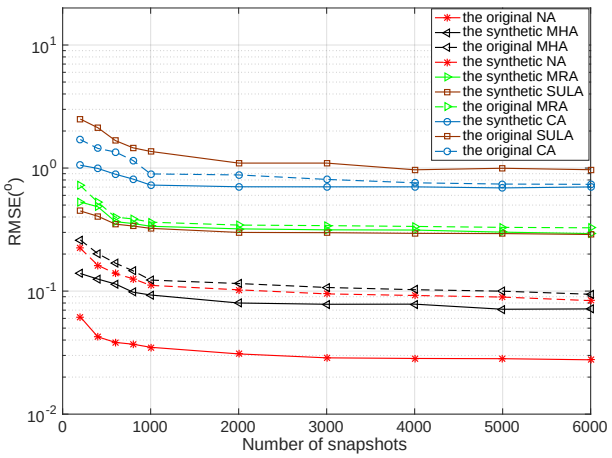



Fig. 11. RMSE vs. snapshots for different sparse arrays using LASSO.

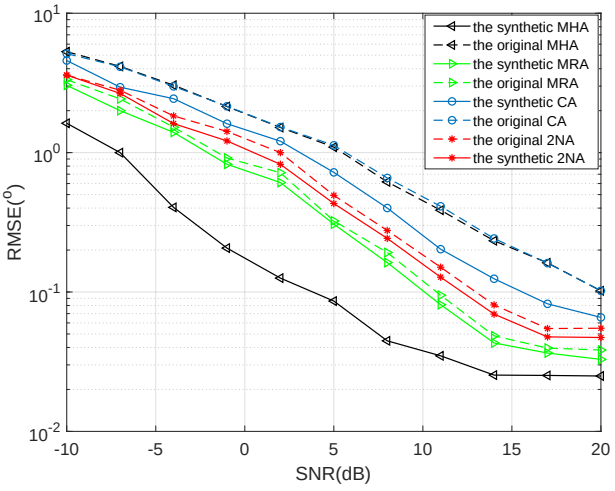



Fig. 12. RMSE vs. SNR for different sparse arrays using MUSIC. (2NA denotes 2-level nested array.)

## VIII. Conclusion

A DOA estimation approach for one-dimensional sparse array moving platform was presented in this paper. By shifting the physical sparse array by half wavelength along its axis, the difference co-array of the combined two array positions consists of the difference co-array of the original array and its unit lag shifted versions in and opposite to direction of motion. The impact of this shifting is filling the majority or all the holes in EISAs. We assessed the rise in DOFs due to array motion. It was shown that the SULAs always yields no-hole co-arrays, whereas the $K$-level nested array obtains the highest number of unique lags. We evaluated the effect of array motion on the number of consecutive lags, and showed that the MHAs obtain the highest number of consecutive lags. These results contribute towards better understanding of the offerings and limitations of the EISAs when considering a moving platform. Supporting simulation examples were provided for DOA estimation utilizing the overall degrees of freedom. These examples demonstrate that EISAs can achieve more

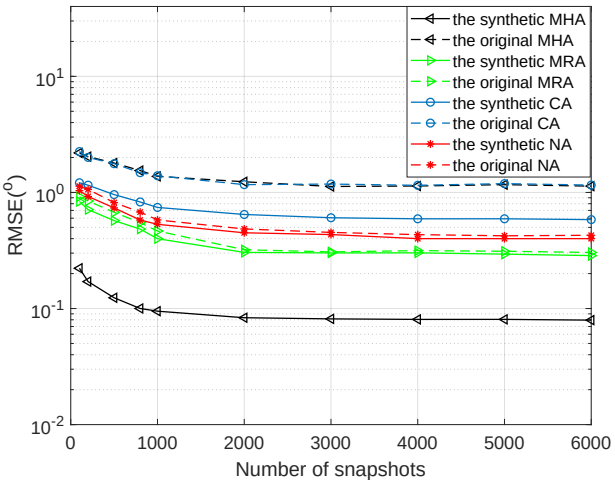



Fig. 13. RMSE vs. snapshots for different sparse arrays using MUSIC.

DOFs and consecutive lags upon a half wavelength array motion. The estimation performance depends on the number of unique lags at the original and the shifted array positions.

## APPENDIX A
## PROOF OF LEMMA 1

The set $\mathbb{S}_{11'} \cup \mathbb{S}_{22'}$ can be rewritten as

$$\begin{aligned}\mathbb{S}_{11'} \cup \mathbb{S}_{22'} = &\left\{M(k_1 - k_1^{'}) - 1\right\} \cup \left\{M(k_1^{'} - k_1) + 1\right\} \\ &\cup \left\{N(k_2 - k_2^{'}) - 1\right\} \cup \left\{N(k_2^{'} - k_2) + 1\right\},\end{aligned} \tag{54}$$

where $-(N-1) \le k_1 - k_1^{'} \le N-1$,$-(M-1) \le k_2 - k_2^{'} \le M-1$.The set $\mathbb{S}_{12'} \cup \mathbb{S}_{21'}$ can be rewritten as

$$\begin{aligned}\mathbb{S}_{12'} \cup \mathbb{S}_{21'} = &\left\{Mk_1 - Nk_2^{'} - 1\right\} \cup \left\{Nk_2 - Mk_1^{'} - 1\right\} \\ &\cup \left\{Nk_2^{'} - Mk_1 + 1\right\} \cup \left\{Mk_1^{'} - Nk_2 + 1\right\}.\end{aligned} \tag{55}$$

When $k_2 = k_2^{'} = 0$ , the set $\mathbb{S}_{12'} \cup \mathbb{S}_{21'}$ can be expressed as

$$\begin{aligned}\mathbb{S}_{12'} \cup \mathbb{S}_{21'} = &\{Mk_1 - 1\} \cup \left\{-Mk_1^{'} - 1\right\} \\ &\cup \{-Mk_1 + 1\} \cup \left\{Mk_1^{'} + 1\right\}.\end{aligned} \tag{56}$$

Because $0 \le k_1^{'}, k_1 \le N-1$, then,

$$\begin{aligned}\left\{M(k_1 - k_1^{'}) - 1\right\} &= \{Mk_1 - 1\} \cup \left\{-Mk_1^{'} - 1\right\} \\ \left\{M(k_1 - k_1^{'}) + 1\right\} &= \{-Mk_1 + 1\} \cup \left\{Mk_1^{'} + 1\right\},\end{aligned} \tag{57}$$

$\mathbb{S}_{12'} \cup \mathbb{S}_{21'}$ can be written as following equation when $k_2 = k_2^{'} = 0$ ,

$$\mathbb{S}_{12'} \cup \mathbb{S}_{21'} = \left\{M(k_1 - k_1^{'}) - 1\right\} \cup \left\{M(k_1^{'} - k_1) + 1\right\}. \tag{58}$$

Similarly, when $k_1 = k_1^{'} = 0$, $\mathbb{S}_{12'} \cup \mathbb{S}_{21'}$ can be expressed as

$$\mathbb{S}_{12'} \cup \mathbb{S}_{21'} = \left\{N(k_2 - k_2^{'}) - 1\right\} \cup \left\{N(k_2^{'} - k_2) + 1\right\}. \tag{59}$$

Based on (58) and (59), it is clear that $\mathbb{S}_{11'} \cup \mathbb{S}_{22'}$ forms a subset of $\mathbb{S}_{12'} \cup \mathbb{S}_{21'}$. Therefore, Lemma 1 is proved.

## APPENDIX B
## PROOF OF LEMMA 2

From the subsection A of the Section IV, we know that additional 2 lags are obtained besides the holes filled after array motion. One is the maximum number of the difference co-array of original array, the other is the minimum number. Therefore, (37) is proved. (38) is the result of equations (32)–(35). Next we solve these equations one by one.

### A. *For equation* (32)

When positive filled holes exist, we have

$$Ml_1^{'} - Nl_2^{'} - 1 = Ml_1 + Nl_2, \tag{60}$$

From (60), we can obtain

$$M(l_1^{'} - l_1) = N(l_2^{'} + l_2) + 1. \tag{61}$$

Applying (31) and (36) to the two sides of (61), we have following two inequations

$$\begin{aligned}2 \le l_1^{'} - l_1 \le N - 2, \\ 1 \le l_2^{'} + l_2 \le (M(N-2) - 1)/N.\end{aligned} \tag{62}$$

Due to $1 \le l_1 \le \text{fix}(N - 1 - N/M)$ , we can obtain the variation range of variable $l_1^{'}$ as $3 \le l_1^{'} \le N-1$. Applying (31) and (36) to the two sides of (60), we have

$$M + N \le Ml_1^{'} - Nl_2^{'} - 1 \le M(N-1) - 1. \tag{63}$$

From (63) and (62), we can obtain

$$\begin{aligned}0 \le l_2^{'} \le M - 1 - (2M+1)/N, \\ 1 \le l_2 \le (M(N-2) - 1)/N.\end{aligned} \tag{64}$$

Rearranging (60), we have the following expressions,

$$l_1^{'} = l_1 + l_2 + l_2^{'} + \frac{(N-M)(l_2 + l_2^{'}) + 1}{M}. \tag{65}$$

It is difficult to solve a underdetermined equation. We give the results under different relations of *M* and *N*. The number of filled holes is given as:

$$\begin{cases} 0, & N = M+1, \\ (N-3), & N = 2M-1, \\ (N - 1 - \kappa_{1+})\alpha_{1+}, & \text{otherwise}, \end{cases}$$

where $\kappa_{1+} = l_2 + l_2^{'} + ((N-M)(l_2 + l_2^{'}) + 1)/M$ is an integer. Integer $\alpha_{1+}$ is the number of combinations of $l_2$,$l_2^{'}$ to make $((N-M)(l_2 + l_2^{'}) + 1)$ divisible by *M*.

Similarly, when the negative filled holes exist, we have the following expression,

$$l_2^{'} = l_2 + l_1 + l_1^{'} - \frac{(N-M)(l_1 + l_1^{'}) + 1}{N}. \tag{66}$$

The number of filled holes is given as:

$$\begin{cases} 0, & N = M+1, \\ (N-3), & N = 2M-1, \\ 0, & N = \beta M + 1, \beta \ge 2, \\ (N - 1 - \kappa_{1+})\alpha_{1+}, & \text{otherwise}, \end{cases}$$

where integer $\kappa_{1-} = l_1 + l_1^{'} - ((N-M)(l_1+l_1^{'})+1)/N$, $\alpha_{1-}$ is the number of combinations of $l_1$,$l_1^{'}$ to make $((N-M)(l_1+l_1^{'})+1)$ divisible by *N*.

From the above, we have a conclusion for (32). The number of holes which are filled due to array motion can be expressed as:

$$\begin{cases} 0, & N = M+1, \\ 2(N-3), & N = 2M-1, \\ (N-1-\kappa_{1+})\alpha_{1+} + (M-1-\kappa_{1-})\alpha_{1-}, & 2M-1 > N > M+1, \\ (N-1-\kappa_{1+})\alpha_{1+}, & N = 2M+1, \\ (N-1-\kappa_{1+})\alpha_{1+} + (M-1-\kappa_{1-})\alpha_{1-}, & N = \beta M-1,\ \beta \geq 3, \\ (N-1-\kappa_{1+})\alpha_{1+}, & N = \beta M+1,\ \beta \geq 3, \\ (N-1-\kappa_{1+})\alpha_{1+} + (M-1-\kappa_{1-})\alpha_{1-}, & N > 2M+1,\ N \neq \beta M \pm 1.\ \beta \geq 3. \end{cases}$$

### *B. For equation* (33)

The derivation process for (33) is similar to that for (32). The number of filled holes is given as:

$$\begin{cases} 2(M-2), & N = M+1, \\ (N-1-\kappa_{2-})\alpha_{2-}, & N = 2M-1, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-1-\kappa_{2-})\alpha_{2-} - 1, & 2M-1 > N > M+1, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-4)\alpha_{2-}, & N = 2M+1, \\ (N-1-\kappa_{2-})\alpha_{2-}, & N = \beta M-1,\ \beta \geq 3, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-1-\kappa_{2-})\alpha_{2-} - 1, & N = \beta M+1,\ \beta \geq 3, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-1-\kappa_{2-})\alpha_{2-} - 1, & N > 2M+1,\ N \neq \beta M \pm 1,\ \beta \geq 3. \end{cases}$$

where $\kappa_{2+} = l_1 + l_1^{'} - ((N-M)(l_1+l_1^{'})-1)/N$, $\kappa_{2-} = l_2 + l_2^{'} + ((N-M)(l_2+l_2^{'})-1)/M$ are integers. Integer $\alpha_{2+}$ is the number of combinations of $l_1$ and $l_1^{'}$ to make $(N-M)(l_1+l_1^{'})-1$ divisible by *N*, and integer $\alpha_{2-}$ is the number of combinations of $l_2$ and $l_2^{'}$ to make $((N-M)(l_2+l_2^{'})-1)$ divisible by *M*.

Before combining the conclusions for (32) and (33), note that there still exist some overlapped holes. We obtain the following equations by letting (32) equal to (33),

$$Ml_3^{'} - Nl_4^{'} - 1 = Nl_4 - Ml_3 - 1 = \pm(Ml_1 + Nl_2). \quad (67)$$

Because $M$ and $N$ are coprime, (67) holds when and only when the following equation holds,

$$l_4^{'} + l_4 = M, l_3^{'} + l_3 = N. \quad (68)$$

The number of overlapped holes is given as:

$$\begin{cases} 0, & N = M+1, \\ (N-1-\kappa_{2-})\alpha_{2-}, & N = 2M-1, \\ (N-1-\kappa_{1+})\alpha_{1+} - 1, & N = 2M+1, \\ (N-1-\kappa_{2-})\alpha_{2-} - (\beta-1), & N = \beta M-1,\ \beta \geq 3, \\ (N-1-\kappa_{1+})\alpha_{1+} - (\beta-1), & N = \beta M+1,\ \beta \geq 3, \\ (N-1-\kappa_{1+})(M-1-\kappa_{+}) & \\ +(M-1-\kappa_{1-})(N-1-\kappa_{2-}), & \text{otherwise.} \end{cases}$$

Therefore, the number of filled holes to satisfy (32) and (33) is given as:

$$\begin{cases} 2(M-2), & N = M+1, \\ 2(N-3), & N = 2M-1, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-4)\alpha_{2-} + 1, & N = 2M+1, \\ (N-1-\kappa_{1+})\alpha_{1+} + (M-1-\kappa_{1-})\alpha_{1-} - 1 + \beta - 1, & N = \beta M-1,\ \beta \geq 3, \\ (M-1-\kappa_{2+})\alpha_{2+} + (N-1-\kappa_{2-})\alpha_{2-} - 1 + \beta - 1, & N = \beta M+1,\ \beta \geq 3, \\ (N-1-\kappa_{1+})\alpha_{1+} + (M-1-\kappa_{1-})\alpha_{1-} & \\ +(M-1-\kappa_{2+})\alpha_{2+} + (N-1-\kappa_{2-})\alpha_{2-} & \\ -(N-1-\kappa_{1+})(M-1-\kappa_{2+}) & \\ -(M-1-\kappa_{1-})(N-1-\kappa_{2-}), & \text{otherwise.} \end{cases}$$

### *C. For equations* (34) *and* (35)

By comparing (32) and (33) with (34) and (35), we conclude that, the left sides of (33) and (34) are opposite in sign. That means that the positive holes filled for (34) have the same positions with the negative holes filled for (33). The negative holes filled for (34) have the same positions with the positive holes filled for (33). The same situation happens in (32) and (35). Therefore, the number of holes filled in (32) and (33) is same as that in (34) and (35). However, there exist the same lags between (32) and (35) as well as between (33) and (34), which can be calculated from the following equations,

$$\begin{aligned} &Ml_3^{'} - Nl_4^{'} - 1 = Nl_4 - Ml_3 + 1 = \pm(Ml_1 + Nl_2), \\ &Nl_4^{'} - Ml_3^{'} - 1 = Ml_3 - Nl_4 + 1 = \pm(Ml_1 + Nl_2), \end{aligned} \quad (69)$$

where $l_3, l_3^{'} \in [0, N-1]$ , $l_4, l_4^{'} \in [0, M-1]$.

Similarly, combining (69) and the conclusion of (32) and (33), we can obtain (38).

## APPENDIX C
## PROOF OF LEMMA 3

The first hole position of difference co-array of the synthetic array is the second hole position of the first consecutive three holes of difference co-array of the original array. Due to the symmetry, here we just consider the positive part of the co-array. For set $\tilde{\mathbb{S}}_{12}$, wen can obtain following equations if the

consecutive three holes of difference co-array of the original array exist.

$$\begin{aligned} Ml_1^{'} - Nl_2^{'} + 1 &= Ml_{m1} + Nl_{n1}, \\ Ml_1^{'} - Nl_2^{'} + 2 &= Ml_{m2} + Nl_{n2}, \\ Ml_1^{'} - Nl_2^{'} + 3 &= Ml_{m3} + Nl_{n3}. \end{aligned} \tag{70}$$

where $(l_{mj}, l_{nj}) \in \mathbb{L}, j = 1, 2, 3$. The first hole position of the synthetic array is determined by solving the minimum value of $Ml_{m1} + Nl_{n1}$ in above equation.

(70) can be rewritten as

$$\begin{aligned} l_1{}' - l_{m1} &= ((N-M)(l_2{}' + l_{n1}) - 1)/M + l_2{}' + l_{n1}, \\ l_1^{'} - l_{m2} &= ((N-M)(l_2{}' + l_{n2}) - 2)/M + l_2{}' + l_{n2}, \\ l_1^{'} - l_{m3} &= ((N-M)(l_2{}' + l_{n3}) - 3)/M + l_2{}' + l_{n3}. \end{aligned} \tag{71}$$

Similar to Appendix B, we obtain

$$1 \le l_2^{'} + l_{nj} \le (M(N-2) - 1)/N. \tag{72}$$

and the first hole position $p_h$ is given as:

$$\begin{cases} 3M + N, & N = M + 1, M > 5, \\ M + 3N + 1, & \\ \quad N = 2M - 1, M \ge 5 \ \& \ N = \beta M - 1, \beta \ge 3, M \ge 4, & \\ (\kappa_{\max +} + 1 - \kappa_{h1+})M + (\alpha_{h1+} - (\alpha_{\min +} - 1))\, N + 1, & \\ \qquad 2M - 1 > N > M + 1, M > 7, & \\ (5 + 2(\beta - 2))M + N + 1, & \\ \qquad N = \beta M + 1, \ \beta \ge 2, M \ge 4, & \\ (\kappa_{\max} + 1 - \kappa_{h1+})M + \alpha_{h1+} N + 1, & \\ \qquad N > 2M + 1, N \ne \beta M \pm 1, \beta \ge 3, & \end{cases}$$

where integer $\kappa_{h1+} = ((N-M)(l_2^{'} + l_{n1}) - 1)/M + l_2^{'} + l_{n1}$, $\alpha_{h1+} = l_2^{'} + l_{n1}$, $\kappa_{h2+} = ((N-M)(l_2^{'} + l_{n2}) - 2)/M + l_2^{'} + l_{n2}$, $\alpha_{h2+} = l_2^{'} + l_{n2}$, $\kappa_{h3+} = ((N-M)(l_2^{'} + l_{n3}) - 3)/M + l_2^{'} + l_{n3}$, $\alpha_{h3+} = l_2^{'} + l_{n3}$. And integer $\kappa_{\max +} = \max(\kappa_{h1+}, \kappa_{h2+}, \kappa_{h2+}), \alpha_{\min +} = \min(\alpha_{h1+}, \alpha_{h2+}, \alpha_{h2+})$.

For set $\tilde{\mathbb{S}}_{12}^{-}$,

$$\begin{aligned} l_2^{'} - l_{n1} &= l_1^{'} + l_{m1} - ((N-M)(l_1^{'} + l_{m1}) + 1)/N, \\ l_2^{'} - l_{n2} &= l_1^{'} + l_{m2} - ((N-M)(l_1^{'} + l_{m2}) + 2)/N, \\ l_2^{'} - l_{n3} &= l_1^{'} + l_{m3} - ((N-M)(l_1^{'} + l_{m3}) + 3)/N, \end{aligned} \tag{73}$$

and $p_h$ is given as:

$$\begin{cases} \text{no holes}, & N = M + 1, M > 5, \\ 2M + 3N + 1, & \\ \quad N = 2M - 1, M \ge 5 \ \& \ N = \beta M - 1, \beta \ge 3, M \ge 4, & \\ \alpha_{h1-} M + (\kappa_{\max -} + 1 - \kappa_{h1-})N + 1, & \\ \qquad 2M - 1 > N > M + 1, M > 7, & \\ (6 + 2(\beta - 2))M + N + 1, & \\ \qquad N = \beta M + 1, \ \beta \ge 2, M \ge 4, & \\ (\alpha_{h1-} - \alpha_{\min -} + 1)\, M + (\kappa_{\max -} + 1 - \kappa_{h1-})N + 1, & \\ \qquad N > 2M + 1, N \ne \beta M \pm 1, \beta \ge 3, & \end{cases}$$

where integer $\kappa_{\max -} = \max(\kappa_{h1-}, \kappa_{h2-}, \kappa_{h2-})$, $\alpha_{\min -} = \min(\alpha_{h1-}, \alpha_{h2-}, \alpha_{h2-})$. Integer $\kappa_{h1-} = l_1^{'} + l_{m1} - ((N-M)(l_1^{'} + l_{m1}) + 1)/N$, $\alpha_{h1-} = l_1^{'} + l_{m1}$, $\kappa_{h2-} = l_1^{'} + l_{m2} - ((N-M)(l_1^{'} + l_{m2}) + 2)/N$, $\alpha_{h2-} = l_1^{'} + l_{m2}$, $\kappa_{h3-} = l_1^{'} + l_{m3} - ((N-M)(l_1^{'} + l_{m3}) + 3)/N$, $\alpha_{h3-} = l_1^{'} + l_{m3}$ .

Combining the conclusions of $\tilde{\mathbb{S}}_{12}$ and $\tilde{\mathbb{S}}_{12}^{-}$, we obtain the expressions of Lemma 3.

## REFERENCES


[1] G. Qin, M. G. Amin, and Y. D. Zhang, "Analysis of coprime arrays on moving platform," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, Brighton, UK, May. 2019, accepted.

[2] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 573–586, Feb. 2011.

[3] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4167–4181, Aug. 2010.

[4] P. Pal and P. P. Vaidyanathan, "Multiple level nested array: An efficient geometry for2$q$th order cumulant based array processing," *IEEE Trans. Signal Process.*, vol. 60, no. 3, pp. 1253–1269, Mar. 2012.

[5] C.-L. Liu and P. P. Vaidyanathan, "Super nested arrays: Linear sparse arrays with reduced mutual coupling – part I: Fundamentals," *IEEE Trans. Signal Process.*, vol. 64, no. 15, pp. 3997–4012, Aug. 2016.

[6] C. Liu and P. P. Vaidyanathan, "Remarks on the spatial smoothing step in coarray music," *IEEE Signal Process. Lett.*, vol. 22, no. 9, pp. 1438–1442, Sep. 2015.

[7] Y. D. Zhang, M. G. Amin, and Himed, "Sparsity-based DOA estimation using co-prime arrays," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, Vancouver, CA, May. 2013, pp. 3967–3971.

[8] 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.

[9] Z. Tan and A. Nehorai, "Sparse direction of arrival estimation using co-prime arrays with off-grid targets," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 26–29, Jan. 2014.

[10] A. Moffet, "Minimum-redundancy linear arrays," *IEEE Trans. Antennas Propagat.*, vol. 16, no. 2, pp. 172–175, Mar. 1968.

[11] W. Wang, S. Ren, and Z. Chen, "Unified coprime array with multi-period subarrays for direction-of-arrival estimation," *Digital Signal Process.*, vol. 74, pp. 30 – 42, 2018.

[12] H. Huang, B. Liao, C. Guo, and J. Huang, "Sparse representation based doa estimation using a modified nested linear array," in *2018 IEEE Radar Conference (RadarConf18)*, Apr. 2018, pp. 0919–0922.

[13] 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, 2018.

[14] C. Zhou, Z. Shi, Y. Gu, and X. Shen, "Decom: Doa estimation with combined music for coprime array," in *Proc. International Conference on Wireless Communications and Signal Processing*, Oct. 2013, pp. 1–5.

[15] M. Yang, L. Sun, X. Yuan, and B. Chen, "A new nested MIMO array with increased degrees of freedom and hole-free difference coarray," *IEEE Signal Process. Lett.*, vol. 25, no. 1, pp. 40–44, Jan. 2018.

[16] Y. Gu, C. Zhou, N. A. Goodman, W. Z. Song, and Z. Shi, "Coprime array adaptive beamforming based on compressive sensing virtual array signal," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, Mar. 2016, pp. 2981–2985.

[17] G. Gelli and L. Izzo, "Minimum-redundancy linear arrays for cyclostationarity-based source location," *IEEE Trans. Signal Process.*, vol. 45, no. 10, pp. 2605–2608, Oct. 1997.

[18] K. Adhikari, J. R. Buck, and K. E. Wage, "Extending coprime sensor arrays to achieve the peak side lobe height of a full uniform linear array," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, p. 148, Sep. 2014.

[19] H. M. Ke Sun, Yimin Liu and X. Wang, "Adaptive sparse representation for source localization with gain/phase errors," *Sensors*, vol. 11, no. 5, pp. 4780–4793, May 2011.

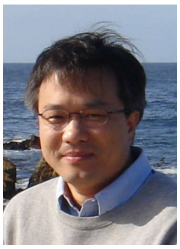
[20] M. Guo, Y. D. Zhang, and T. Chen, "DOA estimation using compressed sparse array," *IEEE Trans. Signal Process.*, vol. 66, no. 15, pp. 4133–4146, Aug. 2018.

[21] 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, pp. 1719–1724(5), August 2017.

[22] X. Yuan, "Coherent source direction-finding using a sparsely-distributed acoustic vector-sensor array," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2710–2715, July 2012.
[23] B. Liao and S. Chan, "Direction-of-arrival estimation in subarrays-based linear sparse arrays with gain/phase uncertainties," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 4, pp. 2268–2280, Oct. 2013.
[24] X. Wang, M. Amin, and X. Cao, "Analysis and design of optimum sparse array configurations for adaptive beamforming," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 340–351, Jan. 2018.
[25] X. Wang, M. Amin, X. Wang, and X. Cao, "Sparse array quiescent beamformer design combining adaptive and deterministic constraints," *IEEE Trans. Antennas Propag.*, vol. 65, no. 11, pp. 5808–5818, Nov. 2017.
[26] E. BouDaher, Y. Jia, F. Ahmad, and M. G. Amin, "Multi-frequency co-prime arrays for high-resolution direction-of-arrival estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3797–3808, Jul. 2015.
[27] J. Ramirez and J. L. Krolik, "Synthetic aperture processing for passive co-prime linear sensor arrays," *Digital Signal Process.*, vol. 61, pp. 62 – 75, 2017.
[28] J. L. Moulton and S. A. Kassam, "Resolving more sources with multi-frequency coarrays in high-resolution direction-of-arrival estimation," in *Annual Conf. Inform. Sciences and Systems*, Mar. 2009, pp. 772–777.
[29] Y. Ming and L. Yi, "Model selection and estimation in regression with grouped variables," *J. Royal Stat. Soc.: Series B*, vol. 68, no. 1, pp. 49–67.
[30] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inform. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.
[31] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal Stat. Soc. Series B*, vol. 58, no. 1, pp. 267–288, 1996.
[32] M. Hawes, L. Mihaylova, F. Septier, and S. Godsill, "Bayesian compressive sensing approaches for direction of arrival estimation with mutual coupling effects," *IEEE Trans. Antennas Propag.*, vol. 65, no. 3, pp. 1357–1368, Mar. 2017.
[33] Z. Yang and L. Xie, "Exact joint sparse frequency recovery via optimization methods," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5145–5157, Oct. 2016.
[34] L. Gan, "Block compressed sensing of natural images," in *Proc. Int. Conf. Digital Signal Process.(DSP2007)*, July 2007, pp. 403–406.
[35] Y. Gu, Y. D. Zhang, and N. A. Goodman, "Optimized compressive sensing-based direction-of-arrival estimation in massive MIMO," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, New Orleans, US, Mar. 2017, pp. 3181–3185.
[36] S. Stergiopoulos and H. Urban, "A new passive synthetic aperture technique for towed arrays," *IEEE J. Oceanic Eng.*, vol. 17, no. 1, pp. 16–25, Jan. 1992.
[37] N. Yen and W. Carey, "Application of synthetic aperture processing to towed-array data," *J. Acoust. Soc. America*, vol. 86, no. 2, pp. 754–764, Aug. 1989.
[38] J. A. Fawcett, "Synthetic aperture processing for a towed array and a moving source," *J. Acoust. Soc. America*, vol. 94, no. 5, pp. 2832–2837, Nov. 1993.
[39] G. Lokourezos, "Prolog to implementation of adaptive and synthetic-aperture processing schemes in integrated active-passive sonar systems," *Proc. IEEE*, vol. 86, no. 2, pp. 356–357, Feb. 1998.
[40] S. Stergiopoulos and E. J. Sullivan, "Extended towed array processing by an overlap correlator," *J. Acoust. Soc. America*, vol. 86, no. 1, pp. 158–171, 1989.
[41] S. Qin, Y. D. Zhang, M. G. Amin, and B. Himed, "DOA estimation exploiting a uniform linear array with multiple co-prime frequencies," *Signal Process.*, vol. 130, pp. 37–46, 2017.
[42] D. A. Linebarger, I. H. Sudborough, and I. G. Tollis, "Difference bases and sparse sensor arrays," *IEEE Trans. Inform. Theory*, vol. 39, no. 2, pp. 716–721, Mar. 1993.

**Guodong Qin** (M'19) received his M.Sc. and Ph.D. degree from Xidian University, China, in 2006 and 2009, in electrical engineering. Since 2010, he has been with the Faculty of the School of Electronic Engineering, Xidian University, China. From 2017 to 2018, he was a Visiting Research Scholar with the Center for Advanced Communications, Villanova University, Villanova, PA, USA. His research interests include direction-of-arrival estimation, sparse array and signal processing, and statistical signal processing.

**Moeness Amin** (F'01) received his B.Sc. degree from Cairo University, Egypt, in 1976, his M.Sc. degree from the University of Petroleum and Minerals, Dhahran, Saudi Arabia, in 1980, and his Ph.D. degree from the University of Colorado, Boulder in 1984, all in electrical engineering. He is director of the Center for Advanced Communications, Villanova University, Pennsylvania. He is the recipient of the 2017 Fulbright Distinguished Chair, the 2016 Alexander von Humboldt Research Award, the 2016 Institution of Engineering and Technology Achievement Medal, the 2014 IEEE Signal Processing Society Technical Achievement Award, the 2009 European Association for Signal Processing Technical Achievement Award, the 2015 IEEE Aerospace and Electronic Systems Society Warren White Award for Excellence in Radar Engineering, and the IEEE Third Millennium Medal. He is a Fellow of the IEEE, the International Society of Optical Engineering, the Institute of Engineering and Technology, and the European Association for Signal Processing.



**Yimin Zhang** (F'19) received the Ph.D. degree from the University of Tsukuba, Tsukuba, Japan, in 1988. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, College of Engineering, Temple University, Philadelphia, PA, USA. From 1998 to 2015, he was a Research Faculty with the Center for Advanced Communications, Villanova University, Villanova, PA, USA. His research interests lie in the areas of statistical signal and array processing, including compressive sensing, machine learning, convex optimization, time-frequency analysis, MIMO systems, radar imaging, direction finding, target localization and tracking, wireless and cooperative networks, and jammer suppression, with applications to radar, wireless communications, and satellite navigation. He has authored more than 350 journal and conference papers and 14 book chapters in these areas.

Dr. Zhang is an Associate Editor for IEEE Transactions on Signal Processing and an Editor for Signal Processing. He was an Associate Editor for IEEE Signal Processing Letters during 20062010, and an Associate Editor for Journal of the Franklin Institute during 20072013. He is a member of the Sensor Array and Multichannel Technical Committee and the Signal Processing Theory and Methods Technical Committee of the IEEE Signal Processing Society. He was the Technical Co-Chair of the 2018 IEEE Sensor Array and Multichannel Signal Processing Workshop. He was the recipient of the 2016 IET Radar, Sonar & Navigation Premium Award and the 2017 IEEE Aerospace and Electronic Systems Society Harry Rowe Mimno Award, and coauthored a paper that received the 2018 IEEE Signal Processing Society Young Author Best Paper Award.