

Article

# Improved 2D Coprime Array Structure with the Difference and Sum Coarray Concept

Guiyu Wang, Zesong Fei, Shiwei Ren  and Xiaoran Li \* 

School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China; guiyu.wang@bit.edu.cn (G.W.); feizesong@bit.edu.cn (Z.F.); renshiwei@bit.edu.cn (S.R.)

\* Correspondence: xiaoran.li@bit.edu.cn; Tel.: +86-010-6891-8356

Received: 14 January 2020; Accepted: 2 February 2020; Published: 5 February 2020



**Abstract:** Recently, the difference and sum (diff-sum) coarray has attracted much attention in one-dimensional direction-of-arrival estimation for its high degrees-of-freedom (DOFs). In this paper, we utilize both the spatial information and the temporal information to construct the diff-sum coarray for planar sparse arrays. The diff-sum coarray contains both the difference coarray and the sum coarray, which provides much higher DOFs than the difference coarray alone. We take a planar coprime array consisting of two uniform square subarrays as the array model. To fully use the aperture-extending ability of the diff-sum coarray, we propose two novel configurations to improve the planar coprime array. The first configuration compresses the inter-element spacing of one subarray and results in a larger consecutive area in the coarray. The second configuration rearranges the two subarrays and introduces a proper separation between them, which can significantly reduce the redundancy of the diff-sum coarray and increase the DOFs. Besides, we derive the closed-form expressions of the central consecutive ranges in the coarrays of the proposed array configurations. Simulations verify the superiority of the proposed array configurations.

**Keywords:** planar coprime array; virtual array; degrees of freedom; direction-of-arrival estimation

## 1. Introduction

Two-dimensional (2D) direction-of-arrival (DOA) estimation has attracted much attention recently for its wide applications in the fields of radar, sonar, communications and navigation, etc. Conventional research mainly focuses on the uniform rectangular arrays (URAs) [1,2], whose inter-element spacing between the adjacent sensors is no more than the typical half-wavelength  $\lambda/2$ . Although the URAs are easy to construct, the detection ability is limited by the number of sensors. The increasing sensor number brings higher hardware cost and computational complexity. Thus, it is significant to design arrays with sparse geometries and the ability to detect more sources than physical sensors.

In the past five years, numbers of novel one-dimensional (1D) sparse arrays with the concept of a coarray have been proposed to achieve a higher number of degrees-of-freedom (DOFs) than ULA. One of the most discussed sparse array configurations, named the nested array [3], is proposed to resolve  $O(N^2)$  sources with  $N$  sensors. The nested array has closed-form expressions of sensor locations and the DOFs. To further enhance the DOFs or to reduce the mutual coupling effects, a series of improved nested array configurations are proposed, such as the extended nested array [4], the augmented nested array [5], and the super nested arrays [6,7]. Another attractive sparse array configuration, named the coprime array [8,9], can resolve  $O(MN)$  sources with  $M + N - 1$  physical sensors, where  $M, N$  are a pair of coprime integer numbers. In order to extend the DOFs or to solve the high peak side lobe problems of the coprime array, various optimized geometries are proposed recently, such as the generalized coprime arrays [10], extended coprime arrays [11,12], and the multi-period coprime structures [13]. With these sparse array configurations, one can construct the difference

coarray by applying the concept of the Khatri–Rao (KR) product to sparse arrays [3–5,8–13]. Finally, under-determined DOA estimation methods, such as the spatial smoothing based MUSIC (SS-MUSIC) method [3,14], can be applied to the resulting difference coarray to detect more sources.

Inspired by the 1D linear sparse arrays introduced above, many planar sparse array configurations have been proposed for 2D virtual aperture enhancement. The L-shaped arrays [15–17] have successfully overcome the ambiguity problem by comparing the results of the sparse subarrays and using fast spectral search methods. However, the number of detectable sources is not significantly increased. In order to obtain a hole-free virtual URA with higher DOFs, several geometries based on the difference coarray are proposed, such as the billboard arrays [18] and open-box arrays (OBA) [19]. Although the OBA geometry can effectively achieve high virtual DOFs, the mutual coupling effects brought by the dense sensor distribution on the boundaries can not be ignored. In order to reduce the mutual coupling effects, a series of improved array structures, for instance the partially open box arrays (POBA), half open box arrays (HOBA), half open box arrays with two layers (HOBA-2), and hourglass arrays [20,21] are proposed one after another. Among the optimized OBAs, the recently proposed half H array (HHA) and ladder array (LA) [22] can achieve the lowest DOA estimation error. On the other hand, a new class of 2D arrays with sensors on the lattice, named 2D nested arrays [23,24], is proposed, which can generate a hole-free difference coarray with much higher virtual DOFs. Besides, the concept of a coprime array is also extended to 2D cases due to the capability of reducing the mutual coupling effects and enhancing the resolution. Recently, a coprime planar array geometry consists of two square uniform planar subarrays [25] is utilized in conventional MUSIC algorithm for 2D DOA estimation. Although the spectral search method provides excellent DOA estimation performance, the computational complexity remains high in two dimensions. A generalized coprime planar array with two rectangular uniform subarrays [26] and the corresponding fast 2D DOA estimation approach [27] are proposed to reduce the complexity significantly. However, the above 2D coprime geometries only make use of the physical sensors to estimate DOAs and the superiority of coarray is not considered. To obtain dense coarrays with high virtual DOFs from sparse arrays, the theory of sparse coprime sensing in multiple dimensions [28,29] is proposed to provide basic configurations and applications of the coprime arrays on lattice. However, conventional array configurations are designed mainly based on the concept of difference coarray. In fact, the sum coarray can be utilized jointly with the difference coarray to further increase DOFs.

In this paper, we improve the vectorized conjugate augmented MUSIC (VCAM) algorithm [30,31] and apply it to 2D DOA estimation based on the second-order statistics. Firstly, we utilize both the spatial information and the temporal information to construct a conjugate augmented correlation vector. By vectorizing the covariance matrix of this vector, we can obtain a difference and sum (diff-sum) coarray, which make use of the complementary between the difference coarray and the sum coarray to construct a larger hole-free area. Based on the diff-sum coarray concept, we then propose two novel 2D coprime configurations. For the first configuration, the inter-element spacing of one subarray is compressed by a positive integer, which results in an array configuration named the coprime array with a compressed subarray (CAACS). The CAACS can acquire much higher DOFs than the prototype 2D coprime array as some of the holes in coarray can be filled up. For the second configuration named as coprime array with two separated subarrays (CATSS), sensors of the two subarrays are rearranged to be bilaterally symmetrical and a proper separation is introduced between the two subarrays. As a result, the diff-sum coarray is redistributed to constitute a central hole-free URA with significantly large number of contiguous virtual sensors. Simulations verify the effectiveness of our proposed methods on increasing the DOFs and improving the DOA estimation performance.

The rest of this paper is organized as follows. In Section 2, the 2D data model and the concept of difference coarray are overviewed. Then, the structure of prototype planar coprime array is introduced. In Section 3, the 2D VCAM algorithm is proposed to generate the diff-sum coarray. Two 2D coprime configurations, i.e., CAACS and CATSS, are introduced in Sections 4 and 5. Simulations provided in Section 6 compare the DOA estimation performance of our proposed array configurations with other

conventional 2D arrays based on the proposed 2D VCAM algorithm. The results demonstrate the validity of the proposed configurations. Section 7 concludes the paper.

Notations: We use lower-case (upper-case) bold characters to denote vectors (matrices), and double-line characters to denote sets.  $(\cdot)^T$  implies the transpose, whereas  $(\cdot)^*$  and  $(\cdot)^H$  denotes complex conjugation and complex conjugate transpose of a matrix or vector, respectively.  $vec(\cdot)$  denotes the vectorization operator that turns a matrix into a vector.  $diag(\mathbf{x})$  denotes a diagonal matrix with the elements of  $\mathbf{x}$  as the diagonal elements, whereas  $\det(\mathbf{X})$  denotes the determinant of a matrix  $\mathbf{X}$ .  $\otimes$  implies the Kronecker product and  $\odot$  implies the KR product. We use  $\times$  between two value ranges to denote the 2D value range, and  $(\cdot)^2$  after a value range to denote that the ranges are the same in two dimensions.

## 2. Preliminaries

### 2.1. Data Model

Suppose  $D$  narrowband, far-field, uncorrelated deterministic plane wave sources with power of  $\sigma_i^2$  for  $i = 1, 2, \dots, D$  impinging to a planar array. Notice that the multipath effect is not considered. The planar array consists of  $N$  sensors located at  $\mathbf{p}_j d$ , where  $\mathbf{p}_j = (p_{xj}, p_{yj}) \in \mathbb{Z}^2$  for  $j = 1, 2, \dots, N$  is an integer vector forming the sensor locations set  $\mathbb{P}$ , and  $d = \lambda/2$  is the unit interval between sensors with  $\lambda$  being the wavelength.  $\mathbf{p}_1$  denotes the reference of array and is set to be the origin  $(0,0)$ . The  $i$ th signal ( $1 \leq i \leq D$ ) can be represented as  $s_i(t) = u_i e^{j\omega_i t}$  where  $u_i$  denotes the deterministic complex amplitude, and  $\omega_i$  is a small frequency offset [32]. The received signal at the  $j$ th sensor can be expressed as:

$$x_j(t) = \sum_{i=1}^D a_{i,j} s_i(t) + n_j(t), \tag{1}$$

where  $a_{i,j} = e^{j\pi(p_{xj} \cos \theta_i \sin \phi_i + p_{yj} \sin \theta_i \sin \phi_i)}$  with  $\theta_i$  and  $\phi_i$  being the azimuth and elevation angles of the  $i$ th source,  $n_j(t)$  is the additive spatially white Gaussian noise with zero mean and variance  $\sigma_n^2$ . Note that  $s_i(t)$  and  $n_j(t)$  are uncorrelated. Rewriting Equation (1) in vector form, we have:

$$\mathbf{x}(t) = \sum_{i=1}^D \mathbf{a}_i s_i(t) + \mathbf{n}(t) = \mathbf{A} \mathbf{s}(t) + \mathbf{n}(t), \tag{2}$$

where  $\mathbf{x}(t) = [x_1(t), \dots, x_N(t)]^T$ ,  $\mathbf{s}(t) = [s_1(t), \dots, s_D(t)]^T$ ,  $\mathbf{a}_i = [a_{i,1}, \dots, a_{i,N}]^T$ ,  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_D]$  is the array manifold matrix,  $\mathbf{n}(t)$  is the noise vector.

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

$$\mathbf{R}_{\mathbf{xx}} = E[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{A} \mathbf{R}_{\mathbf{ss}} \mathbf{A}^H + \sigma_n^2 \mathbf{I}_N, \tag{3}$$

where  $\mathbf{R}_{\mathbf{ss}} = E[\mathbf{s}(t)\mathbf{s}^H(t)] = \text{diag}([\sigma_1^2, \sigma_2^2, \dots, \sigma_D^2])$ . In practice, the data covariance matrix  $\mathbf{R}_{\mathbf{xx}}$  can be estimated by collecting  $K$  snapshots of the received signal from the planar array, i.e.,

$$\mathbf{R}_{\mathbf{xx}} \approx \hat{\mathbf{R}}_{\mathbf{xx}} = \frac{1}{K} \sum_{t=1}^K \mathbf{x}(t)\mathbf{x}^H(t). \tag{4}$$

Vectorizing  $\mathbf{R}_{\mathbf{xx}}$  in Equation (3) yields

$$\mathbf{z} = \text{vec}(\mathbf{R}_{\mathbf{xx}}) = (\mathbf{A}^* \odot \mathbf{A}) \bar{\mathbf{s}} + \sigma_n^2 \mathbf{1}_N, \tag{5}$$

where  $\bar{\mathbf{s}} = [\sigma_1^2, \sigma_2^2, \dots, \sigma_D^2]^T$ ,  $\mathbf{1}_N = \text{vec}(\mathbf{I}_N)$ , and  $\odot$  represents the Khatri–Rao product. As the  $(jk, i)$ th entry in  $\mathbf{A}^* \odot \mathbf{A}$  has the form  $e^{j\pi((p_{xk} - p_{xj}) \cos \theta_i \sin \phi_i + (p_{yk} - p_{yj}) \sin \theta_i \sin \phi_i)} = e^{j\pi(\mathbf{p}_k - \mathbf{p}_j)[\cos \theta_i \sin \phi_i, \sin \theta_i \sin \phi_i]^T}$ ,

$\mathbf{z}$  can be regarded as an equivalent signal vector received from a virtual difference coarray  $\mathbb{D} = \{\mathbf{p} | \mathbf{p} = \mathbf{p}_k - \mathbf{p}_j\}$ , with sensors located at the difference lags.

### 2.2. Prototype Planar Coprime Array

A prototype planar coprime array (PPCA) is illustrated in Figure 1, where  $M_1$  and  $M_2$  are coprime integers. Without loss of generality, we assume  $M_1 > M_2$ . The PPCA consists of two uniform square subarrays. Subarray  $\mathbb{M}_1$  has  $M_2 \times M_2$  sensors with interval  $M_1d$  while subarray  $\mathbb{M}_2$  has  $M_1 \times M_1$  sensors with interval  $M_2d$ .

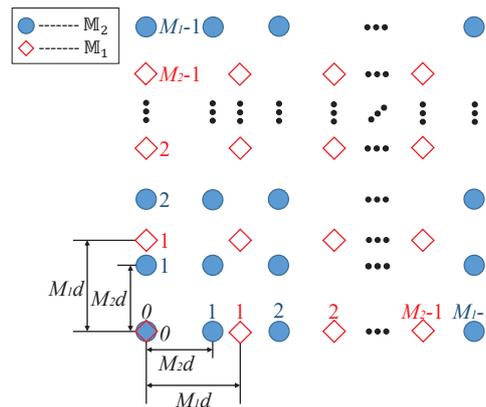


Figure 1. The prototype planar coprime array configuration.

Construct two base matrices  $\mathbf{M}_1 = \begin{bmatrix} M_1 & 0 \\ 0 & M_1 \end{bmatrix}$  and  $\mathbf{M}_2 = \begin{bmatrix} M_2 & 0 \\ 0 & M_2 \end{bmatrix}$ , which generate the base units of the two subarrays. Then, the PPCA sensors are located at

$$\mathbb{P}_{proto} = \{\mathbf{M}_1 \mathbf{n}d\} \cup \{\mathbf{M}_2 \mathbf{m}d\}, \tag{6}$$

where  $\mathbf{n} \in [0 : 1 : M_2 - 1]^2$  and  $\mathbf{m} \in [0 : 1 : M_1 - 1]^2$ . In this paper, we use  $[r_1 : s_1 : r_2]$  to represent the value ranges from  $r_1$  to  $r_2$  by step  $s_1$ . For simplicity, we normalize the unit interval  $d$  to be one in the following representations. As  $M_1$  and  $M_2$  are coprime, the two subarrays share only one overlapped sensor at position  $(0, 0)$ , and the total number of sensors is  $N = M_1^2 + M_2^2 - 1$ .

As there are two subarrays, the difference coarray can be separately considered as the self-difference coarray and the cross-difference coarray, where the cross-difference coarray can be further divided into forward and backward sets. The self-difference coarray, which is the union of the difference sets of the two subarrays themselves, has sensors at

$$\mathbb{L}_{sd} = \mathbb{L}_{sd1} \cup \mathbb{L}_{sd2} = \{\mathbf{M}_1 \mathbf{n}_{sd}\} \cup \{\mathbf{M}_2 \mathbf{m}_{sd}\}, \tag{7}$$

where  $\mathbf{n}_{sd} \in [-(M_2 - 1) : 1 : M_2 - 1]^2$  and  $\mathbf{m}_{sd} \in [-(M_1 - 1) : 1 : M_1 - 1]^2$ . In addition, the forward cross-difference coarray has sensors at

$$\mathbb{L}_{cd}^+ = \{\mathbf{M}_2 \mathbf{m}_{cd} - \mathbf{M}_1 \mathbf{n}_{cd}\}, \tag{8}$$

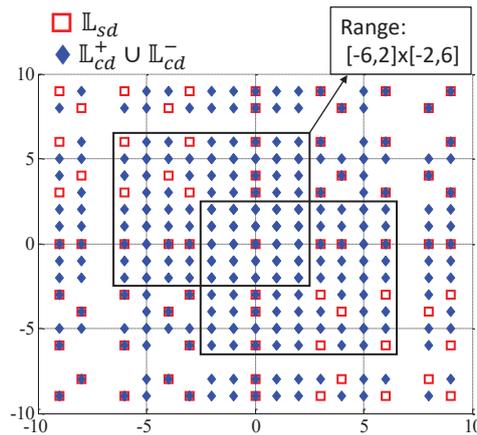
where  $\mathbf{n}_{cd} \in [0 : 1 : M_2 - 1]^2$  and  $\mathbf{m}_{cd} \in [0 : 1 : M_1 - 1]^2$ . The corresponding mirrored backward cross-difference coarray has sensors at  $\mathbb{L}_{cd}^- = \{-\mathbf{l}_{cd} | \mathbf{l}_{cd} \in \mathbb{L}_{cd}^+\}$ . In the consequence, the entire set of sensor positions in the difference coarray is

$$\mathbb{D} = \mathbb{L}_{sd} \cup \mathbb{L}_{cd}^+ \cup \mathbb{L}_{cd}^-. \tag{9}$$

For the sake of evaluating the performance of arrays with the concept of coarray, the definition of DOFs in this paper is established as follows.

**Definition 1** (Degrees of freedom). For a planar array  $\mathbb{P}$  with the concept of coarray, let  $\mathbb{C}$  denotes the corresponding coarray, and  $\mathbb{U}$  denotes the URA with the maximum number of consecutive virtual sensors in  $\mathbb{C}$ . Then, the cardinality of  $\mathbb{C}$  and  $\mathbb{U}$  is called DOFs and uniform DOFs (uDOFs) respectively.

An example of difference coarray of PPCA is illustrated in Figure 2, where  $M_1 = 4$  and  $M_2 = 3$ . The two largest URA areas (which have the same uDOFs) in the difference coarray are marked. The consecutive range is  $\{(x, y) | -6 \leq x \leq 2, -2 \leq y \leq 6\}$  (denoted as  $[-6, 2] \times [-2, 6]$  or  $[-2, 6] \times [-6, 2]$ , and the uDOFs is 81. Note that there are some ‘‘holes’’ in the difference coarray, which limit the uDOFs. A main purpose of this paper is to improve the geometry of PPCA and obtain higher uDOFs.



**Figure 2.** An example of the difference coarray of prototype planar coprime array (PPCA), where  $M_1 = 4$  and  $M_2 = 3$ .

### 3. 2D Vectorized Conjugate Augmented MUSIC Algorithm

In this section, we improve the *Vectorized Conjugate Augmented MUSIC* (VCAM) algorithm [30] and apply it onto 2D DOA estimation.

#### 3.1. The Proposed Method and Coarrays

The first step is to construct the conjugate augmented correlation vector based on the second-order statistics, by utilizing both the temporal and spatial information of the received signals. By collecting  $K$  samples of the first sensor output  $x_1(t)$  and the  $j$ th ( $1 \leq j \leq N$ ) sensor output  $x_j(t)$  with delay  $\tau \neq 0$ , we obtain the sample vectors  $[x_1(1), x_1(2), \dots, x_1(K)]$  and  $[x_j(1 + \tau), x_j(2 + \tau), \dots, x_j(K + \tau)]$  [32]. By calculating the time average function of  $x_1^*(t)$  and  $x_j(t + \tau)$ ,  $j = 1, 2, \dots, N$ , we can obtain

$$\begin{aligned}
 R_{x_1^* x_j}(\tau) &= \frac{1}{K} \sum_{t=1}^K x_1^*(t) x_j(t + \tau) \\
 &= \frac{1}{K} \sum_{t=1}^K \left\{ \left[ \sum_{i=1}^D a_{i,1} s_i(t) + n_1(t) \right]^* \left[ \sum_{l=1}^D a_{l,j} s_l(t + \tau) + n_j(t + \tau) \right] \right\} \\
 &= \frac{1}{K} \sum_{t=1}^K \left\{ \left[ \sum_{i=1}^D \sum_{l=1}^D a_{i,1}^* a_{l,j} s_i^*(t) s_l(t + \tau) + n_1^*(t) n_j(t + \tau) + \sum_{i=1}^D a_{i,1}^* s_i^*(t) n_j(t + \tau) \right. \right. \\
 &\quad \left. \left. + \sum_{l=1}^D a_{l,j} s_l(t + \tau) n_1(t) \right] \right\}.
 \end{aligned} \tag{10}$$

Note that signals and noise are uncorrelated, which means the cross terms between them are 0. Therefore,  $R_{x_1^* x_j}(\tau)$  can be simplified as

$$R_{x_1^* x_j}(\tau) = \sum_{i=1}^D \sum_{l=1}^D \left\{ a_{i,1}^* a_{l,j} \frac{1}{K} \sum_{t=1}^K s_i^*(t) s_l(t + \tau) \right\} + \frac{1}{K} \sum_{t=1}^K n_1^*(t) n_j(t + \tau) \tag{11}$$

$$= \sum_{i=1}^D \sum_{l=1}^D \left\{ a_{i,1}^* a_{l,j} R_{s_i^* s_l}(\tau) \right\} + R_{n_1^* n_j}(\tau),$$

where  $a_{i,1}^* = 1$ ,  $R_{s_i^* s_l}(\tau) = \sum_{t=1}^K s_i^*(t) s_l(t + \tau) / K = u_i^* u_l e^{j\omega_l \tau} \sum_{t=1}^K e^{j(\omega_l - \omega_i)t} / K$  and  $R_{n_1^* n_j}(\tau) = \sum_{t=1}^K n_1^*(t) n_j(t + \tau) / K$ . When  $l \neq i$  and  $K$  is sufficiently large,  $\sum_{t=1}^K e^{j(\omega_l - \omega_i)t} / K \approx 0$ . Besides, as  $n_j(t)$  is zero-mean white Gaussian noise, we have  $R_{n_1^* n_j}(\tau) \approx 0$ . Consequently,  $R_{x_1^* x_j}(\tau)$  can be further simplified as

$$R_{x_1^* x_j}(\tau) = \sum_{i=1}^D a_{i,j} R_{s_i^* s_i}(\tau). \tag{12}$$

where  $R_{s_i^* s_i}(\tau) \approx \sum_{t=1}^K s_i^*(t) s_i(t + \tau) / K = \sigma_i^2 e^{j\omega_i \tau}$ .

Notice that  $R_{s_k^* s_k}(\tau) = \sigma_k^2 e^{j\omega_k \tau}$  has the same form as the source signal, it can be treated as an equivalent signal from the same direction as the original source with power of  $\sigma_k^4$ . By stacking all the vectors  $R_{x_1^* x_j}(\tau)$ , we have

$$\mathbf{v}_{\mathbf{xx}}(\tau) = \mathbf{A} \mathbf{v}_{\mathbf{ss}}(\tau), \tag{13}$$

where  $\mathbf{v}_{\mathbf{xx}}(\tau) = [R_{x_1^* x_1}(\tau), \dots, R_{x_1^* x_N}(\tau)]^T$  and  $\mathbf{v}_{\mathbf{ss}}(\tau) = [R_{s_1^* s_1}(\tau), \dots, R_{s_D^* s_D}(\tau)]^T$ .  $\mathbf{A}$  is the array manifold matrix mentioned before in Section 2.1. Then we invert  $\tau$  to  $-\tau$  and take the conjugate to get

$$[\mathbf{v}_{\mathbf{xx}}(-\tau)]^* = \mathbf{A}^* \mathbf{v}_{\mathbf{ss}}(\tau). \tag{14}$$

The conjugate augmented correlation vector can be constructed by combining Equations (13) and (14) as

$$\mathbf{y}(\tau) = \begin{bmatrix} [\mathbf{v}_{\mathbf{xx}}(-\tau)]^* \\ \mathbf{v}_{\mathbf{xx}}(\tau) \end{bmatrix} = \bar{\mathbf{A}} \mathbf{v}_{\mathbf{ss}}(\tau), \tag{15}$$

where  $\bar{\mathbf{A}} = [\mathbf{A}^*, \mathbf{A}]^T = [\bar{\mathbf{a}}_1, \dots, \bar{\mathbf{a}}_k, \dots, \bar{\mathbf{a}}_D]$  with  $\bar{\mathbf{a}}_k = [\mathbf{a}_k^H, \mathbf{a}_k^T]^T$ .

Suppose that we take  $K_p$  pseudo snapshots with the pseudo sampling interval  $\tau_s$ , which is set to satisfy the sampling theorem, we can construct the pseudo-data matrix of  $\mathbf{y}(\tau)$  as [33]

$$\mathbf{Y} = [\mathbf{y}(\tau_s), \mathbf{y}(2\tau_s), \dots, \mathbf{y}(K_p \tau_s)] = \bar{\mathbf{A}} \mathbf{U} \mathbf{B}, \tag{16}$$

where  $\mathbf{U} = \text{diag}([\sigma_1^2, \dots, \sigma_D^2])$  and  $\mathbf{B} = [\mathbf{b}_1^T, \mathbf{b}_2^T, \dots, \mathbf{b}_D^T]^T$  with  $\mathbf{b}_k = [e^{j\omega_k \tau_s}, e^{j\omega_k 2\tau_s}, \dots, e^{j\omega_k K_p \tau_s}]$ .

Similar to Equation (4) in Section 2, the covariance matrix of  $\mathbf{y}(\tau)$  can be estimated by:

$$\mathbf{R}_{\mathbf{yy}} = \frac{1}{K_p} \mathbf{Y} \mathbf{Y}^H = \bar{\mathbf{A}} \mathbf{U} \left( \frac{1}{K_p} \mathbf{B} \mathbf{B}^H \right) \mathbf{U}^H \bar{\mathbf{A}}^H, \tag{17}$$

where the  $(i, l)$ th entry of the resulting matrix of  $\mathbf{B} \mathbf{B}^H / K_p$  has the form  $\sum_{n=1}^{K_p} e^{j(\omega_i - \omega_l)n\tau_s} / K_p$ . Let  $\bar{\mathbf{R}}_{\mathbf{ss}} = \mathbf{U} (\mathbf{B} \mathbf{B}^H / K_p) \mathbf{U}^H$ , and notice that, when  $l \neq i$  and  $K_p$  is sufficiently large,  $\sum_{n=1}^{K_p} e^{j(\omega_i - \omega_l)n\tau_s} / K_p \approx 0$ . Then we have  $\bar{\mathbf{R}}_{\mathbf{ss}} = \text{diag}([\sigma_1^4, \dots, \sigma_D^4])$ .

Vectorizing  $\mathbf{R}_{yy}$  in Equation (17) yields

$$\mathbf{z} = \text{vec}(\mathbf{R}_{yy}) = (\bar{\mathbf{A}}^* \odot \bar{\mathbf{A}})\bar{\mathbf{s}}, \tag{18}$$

where  $\bar{\mathbf{s}} = [\sigma_1^4, \sigma_2^4, \dots, \sigma_D^4]^T$ . The  $i$ th column of  $\bar{\mathbf{A}}^* \odot \bar{\mathbf{A}}$  has the form

$$\bar{\mathbf{a}}_i^* \otimes \bar{\mathbf{a}}_i = \begin{bmatrix} \mathbf{a}_i^* \\ \mathbf{a}_i \end{bmatrix}^* \otimes \begin{bmatrix} \mathbf{a}_i^* \\ \mathbf{a}_i \end{bmatrix} = \begin{bmatrix} \mathbf{a}_i \otimes \begin{bmatrix} \mathbf{a}_i^* \\ \mathbf{a}_i \end{bmatrix} \\ \mathbf{a}_i^* \otimes \begin{bmatrix} \mathbf{a}_i^* \\ \mathbf{a}_i \end{bmatrix} \end{bmatrix} \tag{19}$$

where  $\mathbf{a}_i^* \otimes \mathbf{a}_i$  is the same as the  $i$ th column of  $\mathbf{A}^* \odot \mathbf{A}$  in Equation (5), while  $\mathbf{a}_i \otimes \mathbf{a}_i^*$  is the conjugate form. They contain the information of difference coarray. As for  $\mathbf{a}_i \otimes \mathbf{a}_i$  and  $\mathbf{a}_i^* \otimes \mathbf{a}_i^*$ , each element has the form of  $e^{\pm j\pi(\mathbf{p}_k + \mathbf{p}_j)[\alpha_i, \beta_i]^T}$  for  $j, k = 1, \dots, N$ , which contains the information of the sum coarray, defined by the following definition:

**Definition 2** (Sum Coarray). Suppose a 2D planar array whose sensors are specified by  $\mathbb{P}$ . The sum coarray  $\mathbb{S}$  consisting of both the positive and the negative summation of each sensor pairs in  $\mathbb{P}$  is defined as

$$\mathbb{S} = \{\pm(\mathbf{p}_j + \mathbf{p}_k) \mid \mathbf{p}_j, \mathbf{p}_k \in \mathbb{P}\}. \tag{20}$$

The sum coarray can be considered as the union of self-sum coarray and cross-sum coarray.

Consequently,  $\mathbf{z}$  behaves like the equivalent signals received from a virtual array with array manifold  $\bar{\mathbf{A}}^* \odot \bar{\mathbf{A}}$ . The virtual array has both the difference and the sum coarray, defined by:

**Definition 3** (Diff-sum Coarray). Suppose a 2D planar array whose sensors are specified by  $\mathbb{P}$ , its diff-sum coarray  $\mathbb{DS}$  is the union of both the difference coarray and the sum coarray:

$$\mathbb{DS} = \mathbb{D} \cup \mathbb{S} = \{\mathbf{p}_j - \mathbf{p}_k\} \cup \{\pm(\mathbf{p}_j + \mathbf{p}_k)\}, \mathbf{p}_j, \mathbf{p}_k \in \mathbb{P}. \tag{21}$$

Similarly, the diff-sum coarray can be divided into the self-diff-sum coarray and the cross-diff-sum coarray.

### 3.2. 2D SS-MUSIC Method

Note that the rank of  $\mathbf{z}$  is one. The subspace based DOA estimation methods, such as MUSIC and ESPRIT, fail to get correct results. In order to solve this problem, we apply the spatial smoothing method [8] to restore the rank. As this method requires a consecutive virtual array, we take average of all the repeated lags in Equation (19) and sort the resulting lags in ascending order. Then, we extract all the consecutive lags of  $\mathbf{z}$  in the range  $[-l_{sx}, l_{sx}] \times [-l_{sy}, l_{sy}]$ . As a result, we obtain a new vector  $\tilde{\mathbf{z}}$  denoted as

$$\tilde{\mathbf{z}} = \tilde{\mathbf{A}}\bar{\mathbf{s}}, \tag{22}$$

where  $\tilde{\mathbf{A}}$  is a  $(2l_{sx} + 1)(2l_{sy} + 1) \times D$  matrix.  $\tilde{\mathbf{A}}$  can be treated as the array manifold of a URA ranging in  $[-l_{sx}, l_{sx}] \times [-l_{sy}, l_{sy}]$ . Divide this virtual URA into  $(l_{sx} + 1)(l_{sy} + 1)$  subarrays,  $\tilde{\mathbf{z}}_{i,j}$ , for  $i = 1, \dots, l_{sx} + 1$  and  $j = 1, \dots, l_{sy} + 1$ , each with  $(l_{sx} + 1)(l_{sy} + 1)$  virtual sensors located at  $[-i + 1, -i + 1 + l_{sx}] \times [-j + 1, -j + 1 + l_{sy}]$ . Finally, we can apply the subspace based DOA estimation methods, like MUSIC and ESPRIT, to the following full rank covariance matrix

$$\mathbf{R}_{zz} = \frac{1}{(l_{sx} + 1)(l_{sy} + 1)} \sum_{j=1}^{l_{sy}+1} \sum_{i=1}^{l_{sx}+1} \tilde{\mathbf{z}}_{i,j} \tilde{\mathbf{z}}_{i,j}^H \tag{23}$$

to estimate the DOAs of the signals.

### 3.3. Complexity Analysis

The computation of the proposed algorithm can be divided into the following three stages:

(1) Construct  $\mathbf{y}(\tau)$  and the corresponding  $\mathbf{R}_{\mathbf{y}\mathbf{y}}$ . For each time delay  $\tau$ ,  $\mathbf{v}_{\mathbf{x}\mathbf{x}}(\tau)$  can be obtained by the time average function with  $K$  snapshots, taking  $\mathcal{O}(NK)$  operations. Thus, the conjugate augmented correlation vector  $\mathbf{y}(\tau)$  needs  $\mathcal{O}(2NK)$  operations to construct. By collecting  $K_p$  pseudo snapshots of  $\mathbf{y}(\tau)$ , the pseudo data matrix  $\mathbf{Y}$  needs  $\mathcal{O}(2NKK_p)$  operations to construct. The total complexity of the corresponding covariance matrix  $\mathbf{R}_{\mathbf{y}\mathbf{y}}$  is  $\mathcal{O}(4N^2K_p)$ . Since we take  $K_p = K$ , the total complexity is  $\mathcal{O}(2NK^2 + 4N^2K)$ .

(2) Establish  $\mathbf{R}_{\mathbf{z}\mathbf{z}}$  by 2D spatial smoothing method. According to Equation (23),  $\mathbf{R}_{\mathbf{z}\mathbf{z}}$  is implemented by the spatial smoothed subarray  $\tilde{\mathbf{z}}_{\mathbf{i},\mathbf{j}}$ . Let  $M_z = l_{sx} + 1$  and  $N_z = l_{sy} + 1$  so that  $\tilde{\mathbf{z}}_{\mathbf{i},\mathbf{j}}$  is a  $M_z \times N_z$  URA. Since each term in Equation (23) takes  $\mathcal{O}(M_z N_z)$  multiplications, the total cost for establishing  $\mathbf{R}_{\mathbf{z}\mathbf{z}}$  is  $\mathcal{O}(M_z^2 N_z^2)$ .

(3) MUSIC spectra. For 2D MUSIC algorithm, since we have obtained the covariance matrix  $\mathbf{R}_{\mathbf{z}\mathbf{z}}$ , the major computational complexity of it is caused by an Eigen-Value Decomposition (EVD) step, and a spectral search step. According to [25], the computational complexity of 2D MUSIC is given as  $\mathcal{O}(M_z^3 N_z^3 + JM_z N_z (M_z N_z - D))$ , where  $J$  denotes the number of spectral points of the total field-of-view.

As a consequence, the total computational complexity of 2D VCAM is  $\mathcal{O}(2NK^2 + 4N^2K + M_z^2 N_z^2 + M_z^3 N_z^3 + JM_z N_z (M_z N_z - D))$ . In the 2D spectral based methods, we usually have  $J \gg N > D$ , which means the complexity of spectral search is much heavier than that of constructing  $\mathbf{R}_{\mathbf{y}\mathbf{y}}$ . Therefore, the complexity of 2D VCAM can be approximately given as  $\mathcal{O}(M_z^3 N_z^3 + JM_z^2 N_z^2)$ .

Table 1 illustrates the computational complexity comparison of different 2D DOA estimation methods applied on PPCA, including the 2D MUSIC algorithm, the total spectral search (TSS) method [25], the partial spectral search (PSS) method [25], and the proposed 2D VCAM. Since  $M_z \times N_z$  refers to the size of the diff-sum coarray, which should be much larger than the PPCA, the complexity of 2D VCAM might be heavier than the other methods. However, the generated virtual array brings much higher uDOFs that can enhance the performance of DOA estimation.

**Table 1.** Computational complexity comparison.

Method	Complexity
2D MUSIC method	$\mathcal{O}(J(M_1^2 + M_2^2)^2)$
TSS in [25]	$\mathcal{O}(JM_1^4 + JM_2^4)$
PSS in [25]	$\mathcal{O}(\frac{J}{M_2^2} M_1^4 + \frac{J}{M_1^2} M_2^4)$
2D VCAM	$\mathcal{O}(M_z^3 N_z^3 + JM_z^2 N_z^2)$

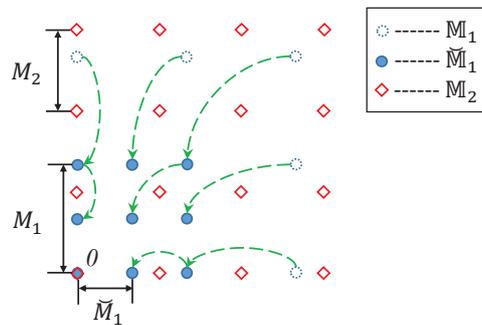
## 4. Coprime Array with a Compressed Subarray

As illustrated in Figure 2, there are many “holes” in the difference coarray of PPCA. To remedy these holes, we propose two novel array configurations to improve increase the consecutive aperture. The first configuration named coprime array with a compressed subarray (CAACS) is presented in this section, whereas the second configuration named coprime array with two separated subarrays (CATSS) is provided in Section 5.

Based on the PPCA geometry, we suppose two subarrays with  $M_1 \times M_1$  and  $M_2 \times M_2$  sensors respectively, where  $M_1$  and  $M_2$  are coprime. Besides, we require either  $M_1$  or  $M_2$  not to be a prime number. Similar with the CACIS configuration [10], we introduce a positive integer compression factor  $p$  to change the inter-element spacing of one subarray. We assume that  $M_1$  can be treated as the product of  $p$  and another positive integer  $\check{M}_1$ , i.e.,

$$M_1 = p\check{M}_1, \tag{24}$$

where  $1 < p \leq M_1$ . It is obvious that  $\check{M}_1$  is also coprime with  $M_2$  since  $M_1$  and  $M_2$  share no common factors other than unity. Figure 3 shows an example of the concept of CAACS configuration, where  $M_1 = 4, M_2 = 3, p = 2, \check{M}_1 = 2$ . It can be seen that, in this array configuration, subarray  $\mathbb{M}_2$  still has  $M_1 \times M_1$  elements with interval of  $M_2$ , whereas subarray  $\check{\mathbb{M}}_1$  has  $M_2 \times M_2$  elements with interval  $\check{M}_1 = M_1/p$ . As a result, the coprime array in this configuration can be considered that the sensor interval of one subarray of the original coprime array is compressed by an integer factor of  $p$ , which leads to the proposed CAACS.



**Figure 3.** The compression process of a coprime array with a compressed subarray (CAACS) configuration, where  $M_1 = 4, M_2 = 3$  and  $p = 2$ .

In this configuration, the self-difference set of the two subarrays becomes

$$\tilde{\mathbb{L}}_{sd} = \tilde{\mathbb{L}}_{sd1} \cup \tilde{\mathbb{L}}_{sd2} = \{\check{\mathbf{M}}_1 \mathbf{n}_{sd}\} \cup \{\mathbf{M}_2 \mathbf{m}_{sd}\}, \tag{25}$$

where  $\mathbf{n}_{sd} \in [-(M_2 - 1) : 1 : M_2 - 1]^2$  and  $\mathbf{m}_{sd} \in [-(M_1 - 1) : 1 : M_1 - 1]^2$ . The forward cross-difference set between the two subarrays becomes

$$\tilde{\mathbb{L}}_{cd}^+ = \{\mathbf{M}_2 \mathbf{m}_{cd} - \check{\mathbf{M}}_1 \mathbf{n}_{cd}\}, \tag{26}$$

where  $\mathbf{n}_{cd} \in [0 : 1 : M_2 - 1]^2$  and  $\mathbf{m}_{cd} \in [0 : 1 : M_1 - 1]^2$ .  $\check{\mathbf{M}}_1 = \begin{bmatrix} \check{M}_1 & 0 \\ 0 & \check{M}_1 \end{bmatrix}$  denotes the base matrix generated by  $\check{M}_1$ .

According to the definition of the sum coarray, the entire set of sensor positions  $\tilde{\mathbb{S}}$  can be represented as

$$\tilde{\mathbb{S}} = \tilde{\mathbb{L}}_{ss}^+ \cup \tilde{\mathbb{L}}_{ss}^- \cup \tilde{\mathbb{L}}_{cs}^+ \cup \tilde{\mathbb{L}}_{cs}^-, \tag{27}$$

where  $\tilde{\mathbb{L}}_{ss}^+$  and  $\tilde{\mathbb{L}}_{ss}^-$  denote the positive and the negative self-sum coarrays, whereas  $\tilde{\mathbb{L}}_{cs}^+$  and  $\tilde{\mathbb{L}}_{cs}^-$  denote the positive and the negative cross-sum coarrays.

The positive self-sum coarray has virtual sensors located at

$$\tilde{\mathbb{L}}_{ss}^+ = \tilde{\mathbb{L}}_{ss1}^+ \cup \tilde{\mathbb{L}}_{ss2}^+ = \{\check{\mathbf{M}}_1 \mathbf{n}_{ss}\} \cup \{\mathbf{M}_2 \mathbf{m}_{ss}\}, \tag{28}$$

where  $\mathbf{n}_{ss} \in [0 : 1 : 2M_2 - 1]^2$  and  $\mathbf{m}_{ss} \in [0 : 1 : 2M_1 - 1]^2$ . In addition, the positive cross-sum coarray has virtual sensors located at

$$\tilde{\mathbb{L}}_{cs}^+ = \{\mathbf{M}_2 \mathbf{m}_{cs} + \check{\mathbf{M}}_1 \mathbf{n}_{cs}\}, \tag{29}$$

where  $\mathbf{n}_{cs} \in [0 : 1 : M_2 - 1]^2$  and  $\mathbf{m}_{cs} \in [0 : 1 : M_1 - 1]^2$ .  $\tilde{\mathbb{L}}_{ss}^-$  and  $\tilde{\mathbb{L}}_{cs}^-$  have sensors located at the mirrored positions of  $\tilde{\mathbb{L}}_{ss}^+$  and  $\tilde{\mathbb{L}}_{cs}^+$ , respectively.

Meanwhile, the corresponding diff-sum coarray can be represented as

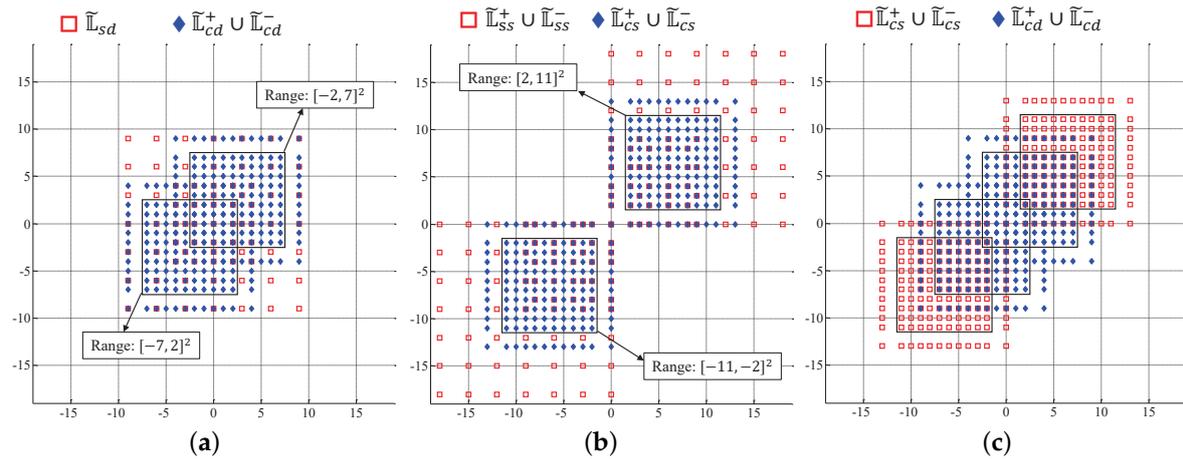
$$\tilde{\mathbb{D}}\tilde{\mathbb{S}} = \tilde{\mathbb{D}} \cup \tilde{\mathbb{S}}. \tag{30}$$

The properties of the difference coarray and the sum coarray are summarized respectively in the following propositions.

**Proposition 1.** *The following facts hold for the difference coarray of CAACS:*

- (a) There are  $\det(\mathbf{M}_1\mathbf{M}_2)$  different 2D integers in the set  $\tilde{\mathbb{L}}_{cd}^+$  and  $\tilde{\mathbb{L}}_{cd}^-$  respectively.
- (b)  $\tilde{\mathbb{L}}_{cd}^+$  contains a URA area that consists of all the consecutive 2D integers in the set  $\{\tilde{\mathbb{I}}_{cd} \mid \tilde{\mathbb{I}}_{cd} \in [-(M_2 - 1), M_1M_2 - \check{M}_1(M_2 - 1) - 1]^2\}$ .  $\tilde{\mathbb{L}}_{cd}^-$  contains a URA located at the mirrored positions.
- (c) The self-difference in the first and third quadrants and that on the X and Y axis form a subset of the cross-difference.
- (d) There are “holes” in the range of  $\tilde{\mathbb{L}}_{cd}^+$  located at  $\tilde{\mathbb{I}}_{cdh}^+ = \{\tilde{\mathbb{I}}_{cdh} = (x, y) \mid x \text{ or } y = -(a\check{M}_1 + bM_2)\} \cup \{\tilde{\mathbb{I}}_{cdh} = (x, y) \mid x \text{ or } y = (a - M_2 + 1)\check{M}_1 + (b + M_1 - 1)M_2\}$ , where  $a \geq 0, b > 0$  are integers.

The proof of Proposition 1 is provided in Appendix A. Figure 4a shows an example of the difference coarray of CAACS with  $M_1 = 4, M_2 = 3, p = 2, \check{M}_1 = 2$ . In this case, the forward cross-difference set consists of 144 unique 2D integers, among which 100 integers in  $[-2, 7]^2$  are consecutive. The backward cross-difference set have the same number of unique 2D integers with consecutive ones in the range  $[-7, 2]^2$ . The two URAs of the cross-difference coarray are marked by solid rectangles. Compared with Figure 2, the uDOFs increase from 81 to 100. We can make use of these enlarged hole-free virtual URAs to execute DOA estimation and get better performance.



**Figure 4.** An example of coarrays of CAACS, where  $M_1 = 4, M_2 = 3$  and  $p = 2$ . (a) The difference coarray. (b) The sum coarray. (c) The cross-diff-sum coarray.

As for the sum coarray of CAACS, it turns out to be:

**Proposition 2.** *The following facts hold for the sum coarray of CAACS:*

- (a) There are  $\det(\mathbf{M}_1\mathbf{M}_2)$  different 2D integers in set  $\tilde{\mathbb{L}}_{cs}^+$  and set  $\tilde{\mathbb{L}}_{cs}^-$ .  $\tilde{\mathbb{L}}_{cs}^+$  and  $\tilde{\mathbb{L}}_{cs}^-$  have only one overlapped point, i.e., the origin  $(0, 0)$ .
- (b)  $\tilde{\mathbb{L}}_{cs}^+$  contains a URA area that consists of all the consecutive 2D integers in set  $\{\tilde{\mathbb{I}}_{cs} \mid \tilde{\mathbb{I}}_{cs} \in [(\check{M}_1 - 1)(M_2 - 1), M_1M_2 - 1]^2\}$ .  $\tilde{\mathbb{L}}_{cs}^-$  contains a URA located at the mirrored positions.
- (c) The positive self-sum coarray contains all the elements of the physical array,  $(\check{M}_1 \cup M_2) \subseteq \tilde{\mathbb{L}}_{ss}^+$ .
- (d) There are “holes” in the range of  $\tilde{\mathbb{L}}_{cs}^+$  located at  $\tilde{\mathbb{I}}_{csh}^+ = \{\tilde{\mathbb{I}}_{csh} = (x, y) \mid x \text{ or } y = a\check{M}_1 + (b + M_1 - 1)M_2\} \cup \{\tilde{\mathbb{I}}_{csh} = (x, y) \mid x \text{ or } y = -(a - M_2 + 1)\check{M}_1 - bM_2\}$ , where  $a \geq 0, b > 0$  are integers.

The proof of Proposition 2 is provided in Appendix B. Figure 4b shows an example of the sum coarray of CAACS with  $M_1 = 4, M_2 = 3, p = 2, \check{M}_1 = 2$ . There are 144 unique virtual sensors in the positive cross-sum coarray and 100 of them are consecutive, ranging in  $[2, 11]^2$ . The negative cross-sum coarray is symmetric with the positive cross-sum coarray about the origin.

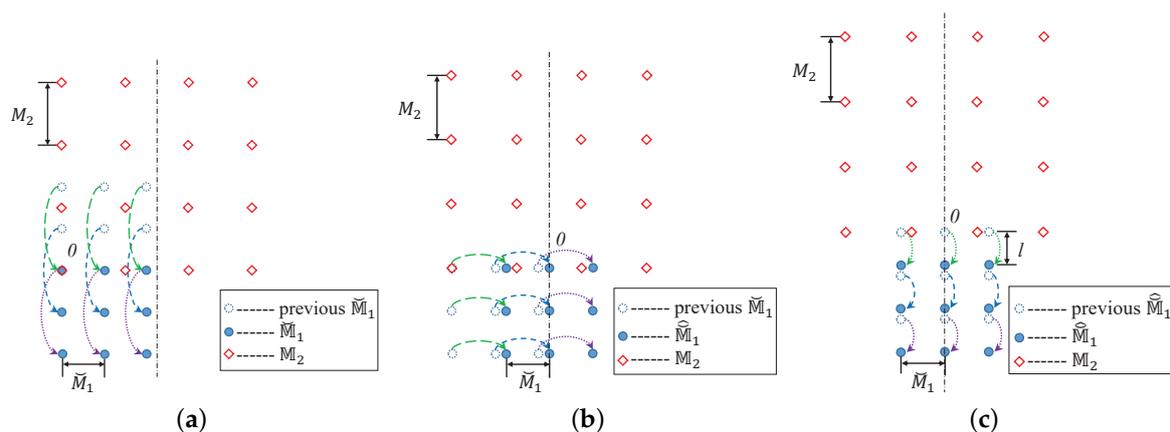
We can draw a conclusion from the above two propositions that for specific coprime integers  $M_1$  and  $M_2$ , smaller values of  $\check{M}_1$  (or in other words, larger values of  $p$ ) result in a larger URA area, which also means higher uDOFs in the coarray. The maximum value that  $p$  can take is  $M_1$ , and the *Sensor Configuration 1* of the 2D nested array structure in [23] is the special case of the CAACS configuration.

**Remark 1.** It can be seen that the self-diff coarray and self-sum coarray make little contribution to increasing the range of the URA area. As mentioned in the previous section, we intend to utilize the contiguous sensors to execute subspace-based DOA estimation, and we only take the cross-coarrays into account in the rest of this paper. Figure 4c illustrates the cross-diff-sum coarray of the CAACS configuration. The cross-diff coarray and the cross-sum coarray have the same construction, in particular, the same size of consecutive areas. Although the URAs in the cross-sum coarray fail to complement the URAs in the cross-diff coarray as we expected, the distribution of the diff-sum coarray of CAACS inspires us with another novel array configuration to utilize those DOFs, which is introduced in detail in Section 5. Note that in the simulation of CAACS, we use the URA in  $\tilde{\mathbb{L}}_{cd}^+$  to conduct DOA estimation.

### 5. Coprime Array with Two Separated Subarrays

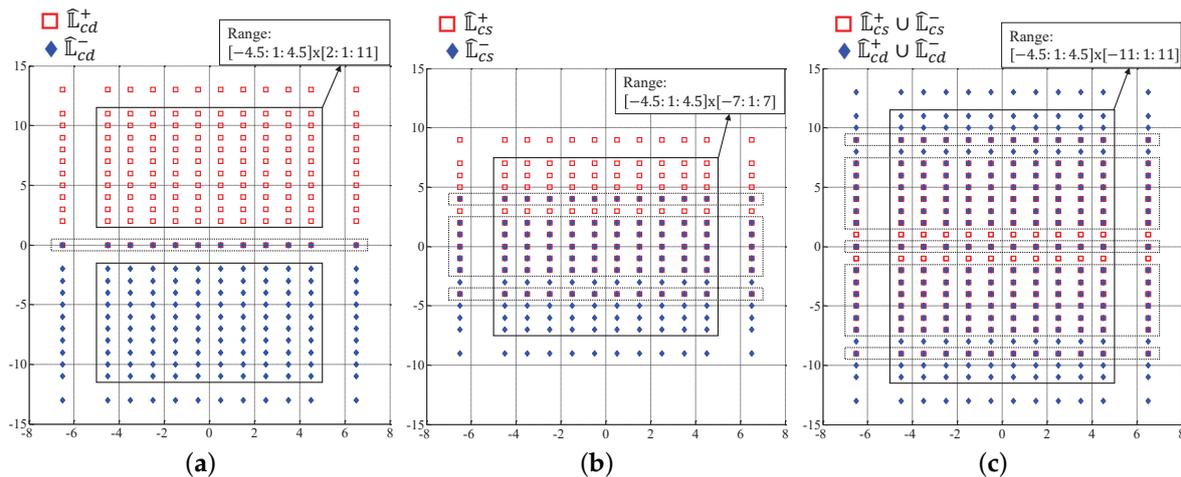
In Section 4 we have found that all the cross-coarrays of CAACS contain hole-free URAs. It is obvious that the diff-sum coarray of CAACS can increase the DOFs and the antenna aperture significantly in comparison with PPCA. However, as we can see in Figure 4c, the cross-coarrays distribute diagonally with only part of their sensors overlapped. Although the total number of consecutive virtual sensors increases, we still fail to make full use of the diff-sum coarray. Note that the four coarrays have the same geometry, which inspires us to rearrange them properly so that an even larger virtual URA can be acquired. This leads to the following array configuration named coprime array with two separated subarrays (CATSS).

Based on the configuration of CAACS, the following steps can further optimize the array geometry. First, all the sensors in  $\check{M}_1$  are moved vertically by  $(M_2 - 1)\check{M}_1$  to make  $\check{M}_1$  and  $M_2$  be on opposite sides of the  $y = 0$  axis as illustrated in Figure 5a. Next, move the sensors of  $\check{M}_1$  horizontally in the positive direction until the entire physical array is bilaterally symmetrical, as demonstrated in Figure 5b. We rebuild the plane coordinate system as follows: (1) let the line where the lower edge of  $M_2$  falls in be the  $y = 0$  axis; (2) let the symmetry axis of  $\check{M}_1$  and  $M_2$  be the  $x = 0$  axis; (3) so that the cross point of them is the new origin  $O$ . From now on, we use  $(\hat{\cdot})$  to denote coarrays of CATSS.



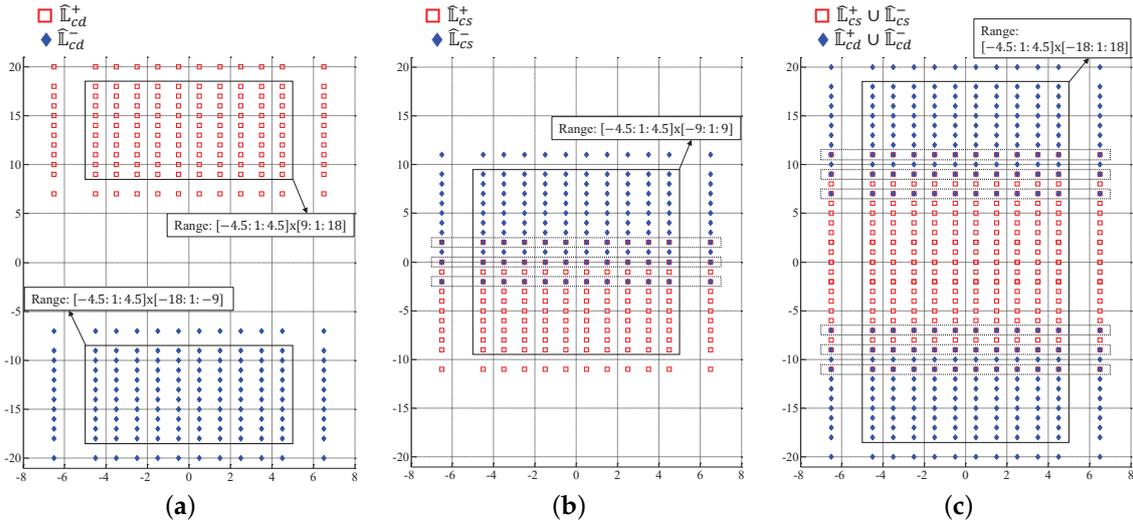
**Figure 5.** Translation process of the coprime array with two separated subarrays (CATSS) configuration, where  $M_1 = 4$ ,  $M_2 = 3$ , and  $p = 2$ . (a) Step 1: Move  $\check{M}_1$  by  $(M_2 - 1)\check{M}_1$  vertically. (b) Step 2: Move  $\check{M}_1$  horizontally to make the whole array bilaterally symmetrical. (c) Step 3: Move  $\check{M}_1$  by  $l$  vertically.

Figure 6 shows an example of the resulting cross coarrays of CATSS. It can be seen that the four cross coarrays are moved to the center of the coordinate plane and are aligned with the same horizontal distribution.  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cd}^-$  locate at the opposite side of the  $y = 0$  axis. The range of the virtual URA in  $\hat{\mathbb{L}}_{cd}^+$  is demonstrated in Figure 6a. Besides, the virtual URAs in  $\hat{\mathbb{L}}_{cs}^+$  and  $\hat{\mathbb{L}}_{cs}^-$  overlap around the  $y = 0$  axis and form a new consecutive area, as shown in Figure 6b. Consequently, the resulting cross-diff-sum coarray contains a central URA which is joined by all four cross coarrays, as illustrated in Figure 6c. The new central URA has 230 contiguous virtual sensors located at  $[-4.5 : 1 : 4.5] \times [-11 : 1 : 11]$ , which is a significant enhancement comparing with the CAACS configuration in Figure 4.



**Figure 6.** Cross coarrays of CATSS, where  $M_1 = 4$ ,  $M_2 = 3$ ,  $p = 2$ , and  $l = 0$ . (a) The cross-difference coarray. (b) The cross-sum coarray. (c) The cross-diff-sum coarray. Dashline boxes mark the overlapped area of the coarrays in each subfigure.

Despite the fact that the cross coarrays have already been joined together to construct a central URA, there is still a large number of overlapped virtual sensors in the URA area. The overlapped areas between  $\hat{\mathbb{L}}_{cs}^+$  and  $\hat{\mathbb{L}}_{cs}^-$  are marked by dashed rectangles in Figure 6b, while the overlapped areas between the cross-difference coarray and the cross-sum coarray are also marked in Figure 6c. By introducing a proper integer separation  $l$  between the two subarrays, the overlap can be reduced so that the DOFs of the resulting cross-diff-sum coarray can be further increased. Specifically, keep  $M_2$  unchanged, while  $M_1$  is shifted along the  $x = 0$  axis by  $l$  in the negative direction. Note that  $0 \leq l \leq M_1 M_2 - (M_2 - 1)M_1 - 1$  must be held to keep the central URA consecutive. Figure 5c shows this operation and Figure 7 illustrates the corresponding cross coarrays with  $M_1 = 4$ ,  $M_2 = 3$ ,  $p = 2$  and the separation  $l = 7$ . As shown in Figure 7a, it is clear that the cross-difference coarray has no overlap between  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cd}^-$ . Meanwhile, the overlap between  $\hat{\mathbb{L}}_{cs}^+$  and  $\hat{\mathbb{L}}_{cs}^-$  has been reduced to only three lines as illustrated in Figure 7b. It can be readily verified that the resulting cross-diff-sum coarray in Figure 7c contains more unique virtual sensors and a much larger central URA in comparison with the cross-diff-sum coarray in Figure 6c. Note that the overlap between cross-difference coarray and cross-sum coarray has been reduced to only six lines.



**Figure 7.** Cross coarrays of CATSS, where  $M_1 = 4$ ,  $M_2 = 3$ ,  $p = 2$  and  $l = 7$ . (a) The cross-difference coarray. (b) The cross-sum coarray. (c) The cross-diff-sum coarray. Dashline boxes mark the overlapped area of the coarrays in each subfigure.

In the CATSS configuration and under the new coordinate system, sensors of the two subarrays are located at

$$\widehat{\mathbf{M}}_1 = \{\check{\mathbf{M}}_1 \mathbf{n} + \mathbf{v}_l\}, \widehat{\mathbf{M}}_2 = \{\mathbf{M}_2 \mathbf{m}\}, \quad (31)$$

where  $\mathbf{n} \in [-0.5(M_2 - 1) : 1 : 0.5(M_2 - 1)] \times [-(M_2 - 1) : 1 : 0]$ ,  $\mathbf{m} \in [-0.5(M_1 - 1) : 1 : 0.5(M_1 - 1)] \times [0 : 1 : M_1 - 1]$  and  $\mathbf{v}_l = (0, -l)^T$ .

Thus, the self-difference set of the two subarrays is

$$\widehat{\mathbf{L}}_{sd} = \widehat{\mathbf{L}}_{sd1} \cup \widehat{\mathbf{L}}_{sd2} = \{\check{\mathbf{M}}_1 \mathbf{n}_{sd}\} \cup \{\mathbf{M}_2 \mathbf{m}_{sd}\}, \quad (32)$$

where  $\mathbf{n}_{sd} \in [-(M_2 - 1) : 1 : M_2 - 1]^2$ ,  $\mathbf{m}_{sd} \in [-(M_1 - 1) : 1 : M_1 - 1]^2$  The forward cross-difference set between the two subarrays becomes

$$\widehat{\mathbf{L}}_{cd}^+ = \{\mathbf{M}_2 \mathbf{m}_{cd} - \check{\mathbf{M}}_1 \mathbf{n}_{cd} - \mathbf{v}_l\}, \quad (33)$$

where  $\mathbf{n}_{cd} \in [-0.5(M_2 - 1) : 1 : 0.5(M_2 - 1)] \times [-(M_2 - 1) : 1 : 0]$  and  $\mathbf{m}_{cd} \in [-0.5(M_1 - 1) : 1 : 0.5(M_1 - 1)] \times [0 : 1 : M_1 - 1]$ .  $\widehat{\mathbf{L}}_{cd}^-$  denotes the mirrored backward cross-diff set. Moreover, the positive self-sum set and the positive cross-sum set are changed into

$$\widehat{\mathbf{L}}_{ss}^+ = \widehat{\mathbf{L}}_{ss1}^+ \cup \widehat{\mathbf{L}}_{ss2}^+ = \{\check{\mathbf{M}}_1 \mathbf{n}_{ss} + 2\mathbf{v}_l\} \cup \{\mathbf{M}_2 \mathbf{m}_{ss}\} \quad (34)$$

and

$$\widehat{\mathbf{L}}_{cs}^+ = \{\mathbf{M}_2 \mathbf{m}_{cs} + \check{\mathbf{M}}_1 \mathbf{n}_{cs} + \mathbf{v}_l\}, \quad (35)$$

where  $\mathbf{n}_{ss} \in [-(M_2 - 1) : 1 : M_2 - 1] \times [-2(M_2 - 1) : 1 : 0]$ ,  $\mathbf{m}_{ss} \in [-(M_1 - 1) : 1 : M_1 - 1] \times [0 : 1 : 2(M_1 - 1)]$ ,  $\mathbf{n}_{cs} \in [-0.5(M_2 - 1) : 1 : 0.5(M_2 - 1)] \times [-(M_2 - 1) : 1 : 0]$ , and  $\mathbf{m}_{cs} \in [-0.5(M_1 - 1) : 1 : 0.5(M_1 - 1)] \times [0 : 1 : M_1 - 1]$ .  $\widehat{\mathbf{L}}_{ss}^-$  is the mirrored negative self-sum set and  $\widehat{\mathbf{L}}_{cs}^-$  is the mirrored cross-sum set.

We also summarize the properties of the difference coarray and the sum coarray of CATSS in the following proposition.

**Proposition 3.** *The following facts hold for CATSS:*

- (a) *There are  $\det(\mathbf{M}_1\mathbf{M}_2)$  different elements in set  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cs}^+$  respectively.*
- (b) *When  $l = 0$ ,  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cd}^-$  have a sole overlapped row on the horizontal axis, whereas when  $l > 0$  they are completely apart.*
- (c)  *$\hat{\mathbb{L}}_{cd}^+$  contains all the consecutive 2D integers in set*

$$\left\{ \hat{\mathbf{I}}_{cd} \mid \hat{\mathbf{I}}_{cd} \in \left[ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 \right] \right. \\ \left. \times \left[ (M_2 - 1)(\check{M}_1 - 1) + l : 1 : M_1M_2 + l - 1 \right] \right\},$$

*while  $\hat{\mathbb{L}}_{cs}^+$  contains all the consecutive 2D integers in set*

$$\left\{ \hat{\mathbf{I}}_{cs} \mid \hat{\mathbf{I}}_{cs} \in \left[ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 \right] \right. \\ \left. \times \left[ -(M_2 - 1) - l : 1 : M_1M_2 - (M_2 - 1)\check{M}_1 - 1 - l \right] \right\}.$$

- (d) *When either  $M_1$  or  $M_2$  is even,  $(\hat{\mathbb{L}}_{sd} \cup \hat{\mathbb{L}}_{ss}^+ \cup \hat{\mathbb{L}}_{ss}^-) \cap (\hat{\mathbb{L}}_{cd}^+ \cup \hat{\mathbb{L}}_{cd}^- \cup \hat{\mathbb{L}}_{cs}^+ \cup \hat{\mathbb{L}}_{cs}^-) = \emptyset$ ; when both  $M_1$  and  $M_2$  are odd,  $(\hat{\mathbb{L}}_{sd} \cup \hat{\mathbb{L}}_{ss}^+ \cup \hat{\mathbb{L}}_{ss}^-)$  is partly overlapped with  $(\hat{\mathbb{L}}_{cd}^+ \cup \hat{\mathbb{L}}_{cd}^- \cup \hat{\mathbb{L}}_{cs}^+ \cup \hat{\mathbb{L}}_{cs}^-)$ .*
- (e) *The holes in the range of  $\hat{\mathbb{L}}_{cd}^+$  are located at*

$$\hat{\mathbb{L}}_{cdh} = \left\{ (x, y) \mid x = \pm \frac{1}{2} [(2a - M_2 + 1)\check{M}_1 + (2b + M_1 - 1)M_2] \right\} \\ \cup \left\{ (x, y) \mid y = -(a - M_2 + 1)\check{M}_1 - bM_2 + l \right\} \cup \left\{ (x, y) \mid y = a\check{M}_1 + (b + M_1 - 1)M_2 + l \right\},$$

*while the holes in the range of  $\hat{\mathbb{L}}_{cs}^+$  are located at*

$$\hat{\mathbb{L}}_{csh} = \left\{ (x, y) \mid x = \pm \frac{1}{2} [(2a - M_2 + 1)\check{M}_1 + (2b + M_1 - 1)M_2] \right\} \\ \cup \left\{ (x, y) \mid y = -a\check{M}_1 - bM_2 - l \right\} \cup \left\{ (x, y) \mid y = (a - M_2 + 1)\check{M}_1 + (b + M_1 - 1)M_2 - l \right\},$$

*where  $a \in [0, M_2 - M_2/\check{M}_1 - 1)$  and  $b \in (0, \check{M}_1(M_2 - 1)/M_2)$  are integers.*

The proof of Proposition 3 is provided in Appendix C. Proposition 3c shows that sensors in all the cross coarrays have the same distribution in the horizontal direction. Proposition 3d and its proof illustrates that sensors in the self coarrays are sparsely distributed in both  $x$  and  $y$  coordinates. Besides, they either fall only  $0.5d$  apart from the cross sensors, or are partly overlapped in the consecutive range. This infers that the self coarrays make no contribution to increase the uDOFs. From Proposition 3e we can deduce that the adjacent cross coarrays can complement their holes with each other. The above facts guarantee that the diff-sum coarray of CATSS contains a hole-free central URA formed by all cross coarrays. Figure 7 provides a direct example of Proposition 3. All four cross coarrays have virtual sensors consecutive in the range  $x \in [-4.5 : 1 : 4.5]$ . Meanwhile,  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cd}^-$  are completely apart while  $\hat{\mathbb{L}}_{cs}^+$  and  $\hat{\mathbb{L}}_{cs}^-$  are continuous with only three lines of overlap. Moreover, it can be found that the lower bound of the  $y$  range in Figure 7b is the upper bound of the  $y$  range of  $\hat{\mathbb{L}}_{cd}^+$  in Figure 7a. This fact means that the URAs in cross-diff coarray and cross-sum coarray can be joined together without any holes. Figure 7c shows the resulting cross-diff-sum coarray, which contains a significantly large central URA with 370 virtual sensors consecutive in  $[-4.5 : 1 : 4.5] \times [-18 : 1 : 18]$ .

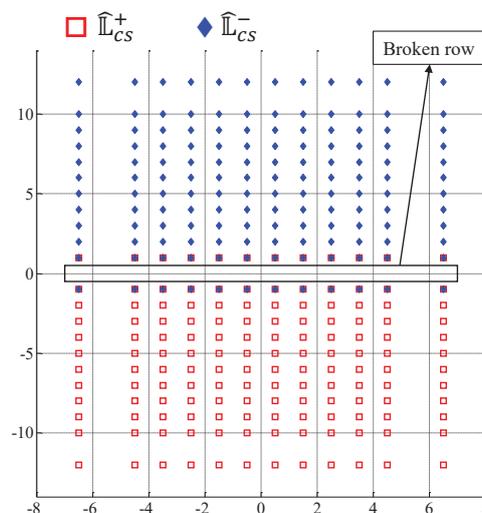
Note that the sensor positions in the vertical direction and the final range of the central URA are relevant to the value of the separation  $l$ , which leads to the following proposition:

**Proposition 4.** The CATSS configuration yields a virtual diff-sum coarray such that:

- (a) Holes below  $\widehat{\mathbb{L}}_{cd}^+$  can be complemented by  $\widehat{\mathbb{L}}_{cs}^-$ , while holes upon  $\widehat{\mathbb{L}}_{cs}^-$  can be complemented by  $\widehat{\mathbb{L}}_{cd}^+$ .
- (b) When integer separation  $l$  takes value in the range  $[0, M_1M_2 - (M_2 - 1)\check{M}_1 - 1]$ ,  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$  form a hole-free URA whose range increases as  $l$  grows. When  $l$  grows to  $M_1M_2 - (M_2 - 1)\check{M}_1$ , a “broken rows” of holes appears at  $y = 0$  axis between  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$ .
- (c) When  $l$  takes a proper value in the range of (b), the cross-diff-sum coarray contains all the consecutive 2D integers in the range

$$\left\{ \widehat{\mathbb{I}}_{DS} \mid \widehat{\mathbb{I}}_{DS} \in \left[ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 \right] \times \left[ -M_1M_2 - l + 1 : 1 : M_1M_2 + l - 1 \right] \right\}.$$

The proof of Proposition 4 is provided in Appendix D. Proposition 4a denotes that the difference coarray and its adjacent sum coarray of CATSS always keep their complementarity so that their consecutive areas can always be joined together. In general, the overlap between two sum coarrays decreases as the separation  $l$  increases, which also means that the final range of the central URA increases. However, when  $l$  exceeds a certain bound, the adjacent sum coarrays would lose their complementarity, and a row of holes would appear between  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$ . Figure 8 gives an example of the property (b) of the above proposition. Compared with Figure 7b, a row of holes appears between  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$  when  $l = 8$ , since the upper bound of  $l$  is 7 in this example.



**Figure 8.** Broken row (or row of holes) appears between  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$ , when  $l = 8 > M_1M_2 - (M_2 - 1)\check{M}_1 - 1 = 7$ .

## 6. Simulation Results

### 6.1. DOF Comparison

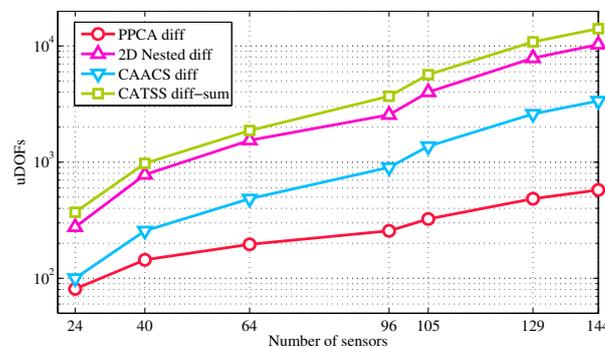
In this sub-section, we provide the comparison of uDOFs of our proposed array configurations and several other 2D array configurations. The selected array configurations are difference coarray of PPCA, difference coarray of 2D nested array Configuration II [23], difference coarray of CAACS, and diff-sum coarray of CATSS.

Table 2 lists the uDOFs expressions of the above array configurations and the generated uDOFs when the number of physical sensors vary from 24 to 144. Figure 9 illustrates the results of the simulation of uDOFs. It can be seen that the diff-sum coarray of CATSS generates the highest uDOFs in all cases, followed by the difference coarray of 2D nested array. This result comes from the fact

that CATSS combines the consecutive part of all difference and sum coarrays, and that 2D nested array combines its positive and negative difference coarrays with only one overlapped line. As the number of sensors increases, the uDOFs of CATSS are significantly enhanced. Note that when the number of sensors is 96, CAACS and CATSS can have two kinds of configurations as the value of compression factor  $p$  varies. We select the configuration that generates the highest uDOFs in the simulation. This simulation verifies the increasing of uDOFs from PPCA to our proposed CAACS and CATSS.

**Table 2.** Uniform degrees of freedom (uDOFs) comparison of different array configurations.

Arrays	uDOFs Expressions	Number of Sensors						
		24	40	64	96	105	129	144
PPCA diff	$(M_1 + 2M_2 - 1)^2$	81	144	196	256	324	484	576
2D Nested diff	$M_1 M_2 (2M_1 M_2 - 1)$	276	780	1540	2556	4005	7875	10,296
CAACS diff	$[M_1 M_2 - (\bar{M}_1 - 1)(M_2 - 1)]^2$	100	256	484	900	1369	2601	3364
CATSS diff-sum	$[M_1 M_2 - (\bar{M}_1 - 1)(M_2 - 1)](2M_1 M_2 + 2l - 1)$	370	976	1870	3690	5661	10,863	14,094

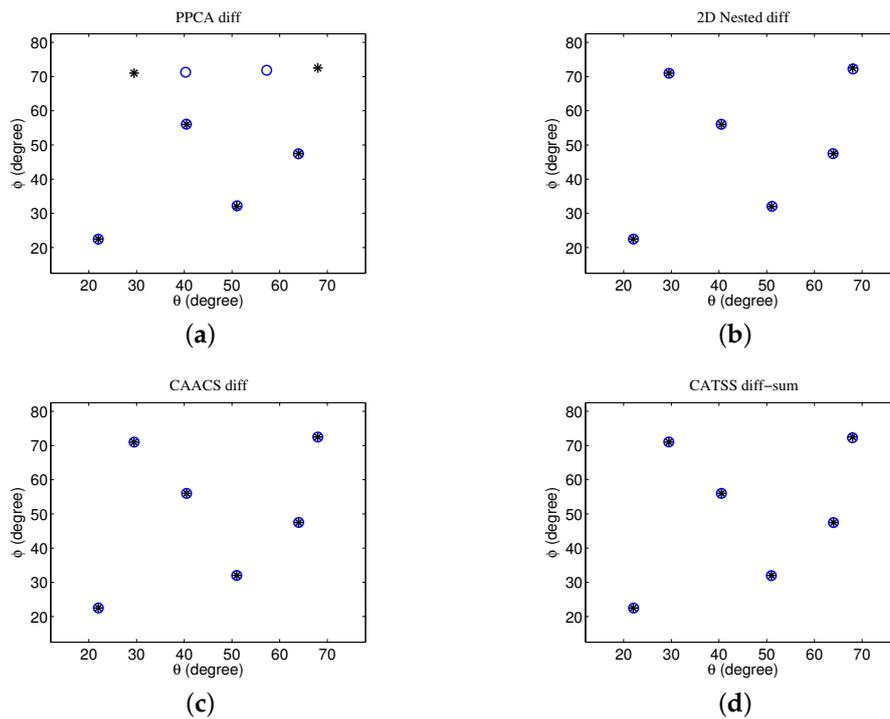


**Figure 9.** Uniform DOFs (uDOFs) comparison of different array configurations.

### 6.2. DOA Estimation

In this sub-section, we take several numerical experiments to demonstrate the superiority our proposed planar coprime array configurations. Assume  $D = 6$  uncorrelated sources with 2D directions distributed at  $(\theta, \phi) \in \{(22^\circ, 22.5^\circ), (29.5^\circ, 71^\circ), (40.5^\circ, 56^\circ), (68^\circ, 72.5^\circ), (51^\circ, 32^\circ), (64^\circ, 47.5^\circ)\}$  impinge onto the PPCA, 2D nested array in [23], CAACS, and CATSS, respectively. For a fair comparison, we set the total number of physical sensors for all array configurations as  $N = 105$ . Thus, the two subarrays are of size  $9 \times 9$  and  $5 \times 5$ . The compression factor for CAACS and CATSS is  $p = 3$  and the separation for CATSS is  $l = 32$ . The corresponding uDOFs for difference coarray of PPCA, difference coarray of 2D nested array, difference coarray of CAACS, and diff-sum coarray of CATSS are 324, 4005, 1369, and 5661 (found in Table 2). Different DOA estimation methods are applied to these array configurations. We use the SS-MUSIC method to work with the difference coarray of PPCA, CAACS, and 2D nested array, while CATSS is with the proposed 2D VCAM algorithm. The searching steps for all spectral searching methods are set to be  $0.1^\circ$ , so that the number of spectral points is  $J = \frac{180^\circ}{0.1^\circ} \times \frac{90^\circ}{0.1^\circ} = 1.62 \times 10^6$ .

Firstly, we compare the DOA estimation results of the six array configurations above. We sample  $K = 200$  snapshots of the received signals from the physical sensors of the arrays, and the pseudo snapshots for 2D VCAM satisfy  $K_p = K$ . The SNR of the signals is set to be 0 dB. We take  $I = 100$  Monte-Carlo trials in these experiments, and Figure 10 shows the average DOA estimation results of these trials. As Figure 10a shows, the difference coarray of PPCA fails to estimate some DOAs due to its low uDOFs. In Figure 10b–d, we can see that all of 2D nested array, CAACS, and CATSS can hit the true DOAs precisely.



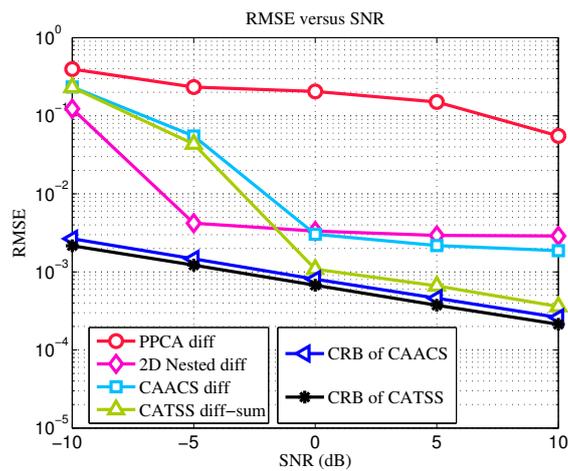
**Figure 10.** The estimated 2D direction-of-arrivals (DOAs) (denoted by  $\circ$ ) and the true 2D DOAs (denoted by  $*$ ) for (a) the difference coarray of prototype planar coprime array (PPCA), (b) the difference coarray of 2D nested array, (c) the difference coarray of CAACS, and (d) the diff-sum coarray of CATSS. The SNR is 0 dB. The number of snapshots is  $K = 200$ .

Next, we study the RMSE performance of these array configurations under different SNRs in the setting with  $K = 200$  snapshots at each SNR. We also study the RMSE performance under different number of snapshots with the SNR fixed as 0 dB. The Cramér–Rao bound (CRB), computed according to formulas provided in [34], is plotted as the benchmark. Note that the RMSE here is defined as

$$RMSE = \sqrt{\frac{1}{IK} \sum_{i=1}^I \sum_{k=1}^K (\hat{\theta}_k(i) - \theta_k)^2 + (\hat{\phi}_k(i) - \phi_k)^2},$$

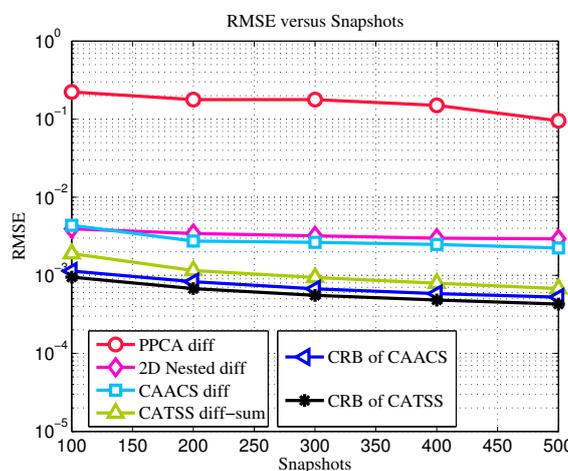
where  $\hat{\theta}_k(i)$  and  $\hat{\phi}_k(i)$  are the estimates of  $\theta_k$  and  $\phi_k$  for the  $i$ th Monte-Carlo trial,  $i = 1, \dots, I$ . Here we take  $I = 100$  independent Monte-Carlo trials for both experiments.

Figure 11 represents the RMSE results of the different array configurations examined at different SNRs. The SNR varies from  $-10$  dB to  $10$  dB. The difference coarray of PPCA performs the worst due to its low uDOFs. In contrast, the 2D nested array and both the proposed CAACS and CATSS outperform the PPCA configuration since their uDOFs are much higher. CAACS and CATSS begin to get better performance than the 2D nested array when the SNR is greater than 0 dB. The performance of the proposed CATSS improves obviously as the SNR gets larger, and is close to its corresponding CRB. Therefore, our proposed array configurations are proved to be effective when the SNR is greater than 0 dB.



**Figure 11.** RMSE as a function of SNR for different array configurations with  $M_1 = 9$ ,  $M_2 = 5$ ,  $p = 3$ ,  $l = 32$ . The number of snapshots is  $K = 200$ , and the pseudo snapshots satisfy  $K_p = K$ .

Figure 12 represents the RMSE results of the different array configurations examined at different numbers of snapshots. The SNR is set to be 0 dB in this experiment. The number of snapshots  $K$  varies from 100 to 500 by a step of 100. The difference coarray of PPCA still fails to obtain satisfying performance even if the snapshots become as large as 500. The RMSE of both CAACS and 2D nested array remains almost the same as  $K$  increases. CAACS performs a little better than 2D nested array when  $K \geq 200$ . Besides, CATSS outperforms all the other configurations and its performance is close to the corresponding CRB. This should be due to the great uDOFs of CATSS as well as both the spatial and temporal information used in the 2D VCAM method. This simulation verifies the superiority of our proposed configuration once again.



**Figure 12.** RMSE as a function of snapshots for different array configurations with  $M_1 = 9$ ,  $M_2 = 5$ ,  $p = 3$ ,  $l = 32$ . The SNR is 0 dB, and the pseudo snapshots satisfy  $K_p = K$ .

**Remark 2.** The simulation results above have verified the superiority of the CATSS configuration with the 2D VCAM method used. It should be noted that the 2D VCAM method might be applied to not only non-cooperative radar systems, but also cooperative settings, to estimate directions of multiple simple pulse radars. In a cooperative radar system, there are  $Q$  tags to be localized, which can be instructed by a central control node to actively transmit mutually orthogonal sinusoidal waveforms with constant amplitude, frequency, and phase. By choosing a set of appropriate frequencies for these signals, the performance of 2D VCAM method can be improved. Thus, a cooperative radar system might make full use of the outstanding performance and high DOFs brought by the 2D VCAM method.

## 7. Conclusions

In this paper, we have utilized the 2D VCAM algorithm to generate a diff-sum coarray and proposed two improved array configurations to enhance the performance of the 2D coprime array, namely compressing the inter-element spacing of one subarray, and introducing a proper separation between two subarrays. The first configuration yields hole-less coarrays with larger consecutive areas, whereas the second configuration rearranges the coarrays to make them fill up the holes of each other and constitute an impressively large consecutive area. The properties of the proposed array configurations are analyzed and the performance of DOA estimation are examined together with the PPCA and 2D nested array. The simulation results shows that both the two operations supply significant improvement to the performance of the array. The CATSS configuration achieves the lowest DOA estimation error and the best performance among all examined array configurations.

**Author Contributions:** Methodology, G.W.; investigation and validation, G.W. and Z.F.; data curation and formal analysis, G.W. and S.R.; supervision, Z.F.; writing—original draft preparation, G.W. and S.R.; writing—review and editing, X.L. All authors have read and agreed to the published version of the manuscript’

**Funding:** This research was supported by the National Natural Science Foundation of China (Grant No. 61801024).

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Proof of Proposition 1

(a) Since the two subarrays  $\check{M}_1$  and  $M_1$  have  $\det(M_1)$  and  $\det(M_2)$  elements respectively, the forward cross-difference coarray  $\tilde{L}_{cd}^+$  consists of  $\det(M_1)\det(M_2) = \det(M_1M_2)$  virtual elements. We then prove it using contradiction. Denote  $\tilde{I}_{cd1} = M_2\mathbf{m}_1 - \check{M}_1\mathbf{n}_1$  and  $\tilde{I}_{cd2} = M_2\mathbf{m}_2 - \check{M}_1\mathbf{n}_2$  as two arbitrary points in set  $\tilde{L}_{cd}^+$ , where  $\mathbf{m}_1 = (m_{1x}, m_{1y}) \in [0 : 1 : M_1 - 1]^2$ ,  $\mathbf{m}_2 = (m_{2x}, m_{2y}) \in [0 : 1 : M_1 - 1]^2$ ,  $\mathbf{n}_1 = (n_{1x}, n_{1y}) \in [0 : 1 : M_2 - 1]^2$ ,  $\mathbf{n}_2 = (n_{2x}, n_{2y}) \in [0 : 1 : M_2 - 1]^2$ . Had  $\tilde{I}_{cd1} = \tilde{I}_{cd2}$  been held, we would have

$$\frac{\check{M}_1}{M_2} = \frac{m_{1x} - m_{2x}}{n_{1x} - n_{2x}} = \frac{m_{1y} - m_{2y}}{n_{1y} - n_{2y}}. \tag{A1}$$

Since  $n_{1x} - n_{2x} < M_2$  and  $n_{1y} - n_{2y} < M_2$ , Equation (A1) cannot be held due to the coprimality of  $\check{M}_1$  and  $M_2$ , i.e.,  $\tilde{I}_{cd1}$  and  $\tilde{I}_{cd2}$  cannot be equal. Thus, there are  $\det(M_1M_2)$  different integer elements in set  $\tilde{L}_{cd}^+$ . For the mirrored backward cross-different coarray  $\tilde{L}_{cd}^-$ , we can have the same conclusion through the similar way.

(b) Given an arbitrary integer point  $\tilde{I}_{cd}$  in set  $\tilde{L}_{cd}^+$  satisfying

$$\tilde{I}_{cd} \in [-(M_2 - 1), M_1M_2 - \check{M}_1(M_2 - 1) - 1]^2, \tag{A2}$$

Our purpose is to prove that there exist integer vectors  $\mathbf{m}_{cd} \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n}_{cd} \in [0 : 1 : M_2 - 1]^2$  such that  $\tilde{I}_{cd} = M_2\mathbf{m}_{cd} - \check{M}_1\mathbf{n}_{cd}$  holds. The requirement  $\mathbf{n}_{cd} \in [0 : 1 : M_2 - 1]^2$  can be rewritten as  $\mathbf{n}_{cd} \in [0, M_2 - 1]^2$ , which is equivalent to

$$\check{M}_1\mathbf{n}_{cd} \in [0, \check{M}_1(M_2 - 1)]^2. \tag{A3}$$

As  $M_2\mathbf{m}_{cd} = \tilde{I}_{cd} + \check{M}_1\mathbf{n}_{cd}$ , the following relationship can be obtained by combing Equations (A2) and (A3),

$$M_2\mathbf{m}_{cd} \in [-(M_2 - 1), M_1M_2 - 1]^2. \tag{A4}$$

This relationship can be equally expressed as  $M_2\mathbf{m}_{cd} \in (-M_2, M_1M_2)^2$ , which implies  $\mathbf{m}_{cd} \in (-1, M_1)^2$ . Because  $\mathbf{m}_{cd}$  is an integer vector, Equation (A4) is equivalent to  $\mathbf{m}_{cd} \in [0, M_1 - 1]^2$ , which is satisfied in  $\tilde{L}_{cd}^+$ .

(c) Because the two subarrays share the first sensor at the origin, the self-differences in the first and third quadrants and that on the X and Y axis can be taken as the cross-differences between every sensor

of one subarray and the first sensor of the other subarray. Thus,  $\{\tilde{\mathbf{I}}_{sd} = (x, y) \mid \tilde{\mathbf{I}}_{sd} \in \tilde{\mathbb{L}}_{sd}, xy \geq 0\} \subseteq \{\tilde{\mathbb{L}}_{cd}^+ \cup \tilde{\mathbb{L}}_{cd}^-\}$ .

(d) We use contradiction to prove the sufficiency of this property. Suppose  $\mathbf{M}_2 \mathbf{m}_{cd} - \check{\mathbf{M}}_1 \mathbf{n}_{cd} \in \tilde{\mathbb{L}}_{cdh}^+$  holds for some integer vectors  $\mathbf{m}_{cd} = (x_1, y_1) \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n}_{cd} = (x_2, y_2) \in [0 : 1 : M_2 - 1]^2$ . When  $M_2 x_1 - \check{M}_1 x_2 = -(a\check{M}_1 + bM_2)$  where  $a \geq 0$  and  $b > 0$ , the relationship

$$\frac{M_2}{\check{M}_1} = \frac{x_2 - a}{x_1 + b} \tag{A5}$$

must be valid. Since  $0 \leq x_2 < M_2$  and  $a \geq 0$ , we have  $x_2 - a < M_2$ . Due to the coprimality between  $\check{M}_1$  and  $M_2$ , there is no such integer  $x_1$  that satisfies Equation (A5). Similarly, when  $M_2 x_1 - \check{M}_1 x_2 = (a - M_2 + 1)\check{M}_1 + (b + M_1 - 1)M_2$ , the relationship

$$\frac{M_2}{\check{M}_1} = \frac{M_2 - (x_2 + a + 1)}{M_1 + b - 1 - x_1} \tag{A6}$$

must be valid. Since  $M_2 - (x_2 + a + 1) < M_2$  and due to the coprimality between  $\check{M}_1$  and  $M_2$ , there is no such integer  $x_1$  that satisfies Equation (A6). Same results can be obtained for the  $y$  coordinate. Therefore, the assumption cannot hold, i.e., there are holes located at  $\tilde{\mathbb{L}}_{cdh}^+$ .

As for the necessity, we need to prove that  $\tilde{\mathbb{L}}_{cdh}^+$  contains all holes within the boundary of  $\tilde{\mathbb{L}}_{cd}^+$ . Equivalently, we need to check the number of elements in  $\tilde{\mathbb{L}}_{cdh}^+$ . From the expression of  $\tilde{\mathbb{L}}_{cdh}^+$  and Proposition 1a, it is easy to know that the total number of holes within the boundary of  $\tilde{\mathbb{L}}_{cd}^+$  is

$$C_{hole} = [M_2(M_1 - 1) + \check{M}_1(M_2 - 1) + 1]^2 - (M_1 M_2)^2. \tag{A7}$$

According to the expression of  $\tilde{\mathbb{L}}_{cdh}^+$  and Proposition 1b, we need to verify how many pairs of  $(a, b)$  exist in the inequation  $-\check{M}_1(M_2 - 1) \leq -(a\check{M}_1 + bM_2) \leq -(M_2 - 1)$  and  $M_1 M_2 - \check{M}_1(M_2 - 1) - 1 \leq (a - M_2 + 1)\check{M}_1 + (b + M_1 - 1)M_2 \leq M_2(M_1 - 1)$ , where  $a \geq 0, b > 0$ . The two inequations above can be simplified to the same form as

$$M_2 - 1 \leq a\check{M}_1 + bM_2 \leq \check{M}_1(M_2 - 1), a \geq 0, b \geq 1. \tag{A8}$$

Combining Equation (A8) and the condition  $b \geq 1$ , we have  $0 \leq a \leq M_2 - 1 - M_2/\check{M}_1$ . Assume that  $\check{M}_1 > M_2$  and then it becomes  $0 \leq a \leq M_2 - 2$ . If  $a = 0$ , we have  $1 \leq b \leq \check{M}_1 - \lfloor \check{M}_1/M_2 \rfloor - 1$ . Here,  $\lfloor \cdot \rfloor$  means the integer no more than the expression inside. Absolutely, there are  $\check{M}_1 - \lfloor \check{M}_1/M_2 \rfloor - 1$  pairs of  $(a, b)$  in this case. If  $a = M_2 - 2$ , we have  $1 \leq b \leq \lfloor \check{M}_1/M_2 \rfloor$  so that there are  $\lfloor \check{M}_1/M_2 \rfloor$  pairs. By putting the above two cases in pair, we can find  $\check{M}_1 - 1$  pairs of  $(a, b)$  satisfy Equation (A8). Similarly,  $a = 1$  and  $a = M_2 - 3$  are also in pair to generate  $\check{M}_1 - 1$  pairs of  $(a, b)$ . When  $M_2 - 1$  is even, there are in total  $(M_2 - 1)(\check{M}_1 - 1)/2$  pairs of  $(a, b)$ . When  $M_2 - 1$  is odd, then  $\check{M}_1$  must be odd. In this case,  $(M_2 - 2)(\check{M}_1 - 1)/2$  pairs of  $(a, b)$  exist with  $a = (M_2 - 2)/2$  left. When  $a = (M_2 - 2)/2$ , we have  $1 \leq b \leq (\check{M}_1 - 1)/2$ . As such, we have in total  $(M_2 - 2)(\check{M}_1 - 1)/2 + (\check{M}_1 - 1)/2 = (M_2 - 1)(\check{M}_1 - 1)/2$  pairs of  $(a, b)$ .

The same result can be easily obtained when  $\check{M}_1 < M_2$  by considering  $a$  we can acquire on condition that  $b$  changes in the range  $[1, \check{M}_1 - 1]$ . In this situation,  $b = \check{M}_1 - 1$  and  $b = 1$  are in pair. The final total number of pairs of  $(a, b)$  can be verified to be  $(M_2 - 2)(\check{M}_1 - 1)/2 + (\check{M}_1 - 1)/2 = (M_2 - 1)(\check{M}_1 - 1)/2$  as well.

Consequently, the total number of holes in  $\tilde{\mathbb{L}}_{cdh}^+$  is four times the number of pairs of  $(a, b)$  above, except for the repeats caused by the cross points of in  $\tilde{\mathbb{L}}_{cdh}^+$ , i.e.,

$$C'_{hole} = 4 \times \frac{(M_2 - 1)(\check{M}_1 - 1)}{2} [M_2(M_1 - 1) + \check{M}_1(M_2 - 1) + 1] - 4 \times \left[ \frac{(M_2 - 1)(\check{M}_1 - 1)}{2} \right]^2. \tag{A9}$$

It can be easily verified that  $C'_{hole}$  above is equivalent to  $C_{hole}$  in Equation (A7) which has proved the necessity of the property.

**Appendix B. Proof of Proposition 2**

(a) Similar to Proposition 1, there are  $\det(\mathbf{M}_1\mathbf{M}_2)$  virtual elements in  $\tilde{\mathbb{L}}_{cs}^+$  and  $\tilde{\mathbb{L}}_{cs}^-$  respectively. We still use contradiction to achieve the following proof. Denote  $\tilde{\mathbf{l}}_{cs1} = \mathbf{M}_2\mathbf{m}_1 + \check{\mathbf{M}}_1\mathbf{n}_1$  and  $\tilde{\mathbf{l}}_{cs2} = \mathbf{M}_2\mathbf{m}_2 + \check{\mathbf{M}}_1\mathbf{n}_2$  as two arbitrary points in set  $\tilde{\mathbb{L}}_{cs}^+$ , where  $\mathbf{m}_1 = (m_{1x}, m_{1y}) \in [0 : 1 : M_1 - 1]^2$ ,  $\mathbf{m}_2 = (m_{2x}, m_{2y}) \in [0 : 1 : M_1 - 1]^2$ ,  $\mathbf{n}_1 = (n_{1x}, n_{1y}) \in [0 : 1 : M_2 - 1]^2$ ,  $\mathbf{n}_2 = (n_{2x}, n_{2y}) \in [0 : 1 : M_2 - 1]^2$ . Had  $\tilde{\mathbf{l}}_{cs1} = \tilde{\mathbf{l}}_{cs2}$  been held, we would have

$$\frac{\check{M}_1}{M_2} = \frac{m_{1x} - m_{2x}}{n_{2x} - n_{1x}} = \frac{m_{1y} - m_{2y}}{n_{2y} - n_{1y}}. \tag{A10}$$

Since  $n_{2x} - n_{1x} < M_2$  and  $n_{2y} - n_{1y} < M_2$ , Equation (A10) cannot be held due to the coprimality of  $\check{M}_1$  and  $M_2$ , i.e.,  $\tilde{\mathbf{l}}_{cs1}$  and  $\tilde{\mathbf{l}}_{cs2}$  cannot be equal. Thus, there are  $\det(\mathbf{M}_1\mathbf{M}_2)$  different integer elements in set  $\tilde{\mathbb{L}}_{cs}^+$ . For the mirrored negative cross-sum coarray  $\tilde{\mathbb{L}}_{cs}^-$ , we can have the same conclusion through the similar way.

Besides, according to the definition of the sum coarray, we can easily find that the 2D coordinates of all points in  $\tilde{\mathbb{L}}_{cs}^+$  are not negative, while the 2D coordinates of all points in  $\tilde{\mathbb{L}}_{cs}^-$  are not positive. Thus,  $\tilde{\mathbb{L}}_{cs}^+$  and  $\tilde{\mathbb{L}}_{cs}^-$  have no overlapped point other than the origin (0, 0).

(b) Given an arbitrary integer point  $\tilde{\mathbf{l}}_{cs}$  in set  $\tilde{\mathbb{L}}_{cs}^+$  satisfying

$$\tilde{\mathbf{l}}_{cd} \in [(\check{M}_1 - 1)(M_2 - 1), M_1M_2 - 1]^2, \tag{A11}$$

We intend to prove that there exist integer vectors  $\mathbf{m}_{cs} \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n}_{cs} \in [0 : 1 : M_2 - 1]^2$  such that  $\tilde{\mathbf{l}}_{cs} = \mathbf{M}_2\mathbf{m}_{cs} + \check{\mathbf{M}}_1\mathbf{n}_{cs}$  holds. The requirement  $\mathbf{n}_{cs} \in [0 : 1 : M_2 - 1]^2$  can be rewritten as  $\mathbf{n}_{cs} \in [0, M_2 - 1]^2$ , which is equivalent to

$$\check{\mathbf{M}}_1\mathbf{n}_{cs} \in [0, \check{M}_1(M_2 - 1)]^2. \tag{A12}$$

As  $\mathbf{M}_2\mathbf{m}_{cs} = \tilde{\mathbf{l}}_{cs} - \check{\mathbf{M}}_1\mathbf{n}_{cs}$ , the following relationship can be obtained by combing Equations (A11) and (A12),

$$\mathbf{M}_2\mathbf{m}_{cs} \in [-(M_2 - 1), M_1M_2 - 1]^2. \tag{A13}$$

This relationship can be equally expressed as  $\mathbf{M}_2\mathbf{m}_{cs} \in (-M_2, M_1M_2)^2$ , which implies  $\mathbf{m}_{cs} \in (-1, M_1)^2$ . Because  $\mathbf{m}_{cs}$  is an integer vector, Equation (A13) is equivalent to  $\mathbf{m}_{cs} \in [0, M_1 - 1]^2$ , which is satisfied in  $\tilde{\mathbb{L}}_{cs}^+$ .

(c) As shown in Equations (6) and (28), the expression of sensor positions of the physical array and that of the positive cross-sum coarray have the same formation, while the ranges of vector  $\mathbf{n}$  and  $\mathbf{m}$  are different. Because  $[0 : 1 : M_2 - 1]^2 \subset [0 : 1 : 2M_2 - 1]^2$  and  $[0 : 1 : M_1 - 1]^2 \subset [0 : 1 : 2M_1 - 1]^2$ , the positive cross-sum set contains all elements in the physical array set, i.e.,  $(\check{\mathbb{M}}_1 \cup \mathbb{M}_2) \subseteq \tilde{\mathbb{L}}_{ss}^+$

(d) We use contradiction to prove the sufficiency. Suppose  $\mathbf{M}_2\mathbf{m}_{cs} + \check{\mathbf{M}}_1\mathbf{n}_{cs} \in \tilde{\mathbb{L}}_{csh}^+$  holds for some integer vectors  $\mathbf{m}_{cs} = (x_1, y_1) \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n}_{cs} = (x_2, y_2) \in [0 : 1 : M_2 - 1]^2$ . When  $M_2x_1 + \check{M}_1x_2 = a\check{M}_1 + (b + M_1 - 1)M_2$  where  $a \geq 0$  and  $b > 0$ , the relationship

$$\frac{M_2}{\check{M}_1} = \frac{x_2 - a}{-x_1 + b + M_1 - 1} \tag{A14}$$

must be valid. Since  $0 \leq x_2 < M_2$  and  $a \geq 0$ , we have  $x_2 - a < M_2$ . Due to the comprimality between  $\check{M}_1$  and  $M_2$ , there is no such integer  $x_1$  that satisfies Equation (A14). Similarly, when  $M_2x_1 + \check{M}_1x_2 = -(a - M_2 + 1)\check{M}_1 - bM_2$ , the relationship

$$\frac{M_2}{\check{M}_1} = \frac{M_2 - (x_2 + a + 1)}{b + x_1} \tag{A15}$$

must be valid. Since  $M_2 - (x_2 + a + 1) < M_2$  and due to the comprimality between  $\check{M}_1$  and  $M_2$ , there is no such integer  $x_1$  that satisfies (A15). Same results can be obtained for the  $y$  coordinate. Therefore, the assumption cannot be hold, i.e., there are holes located at  $\tilde{\mathbb{L}}_{csh}^+$ .

The necessity can be proved in the similar way of Proposition 1d. It is easy to know that the total number of holes within the boundary of  $\tilde{\mathbb{L}}_{cs}^+$  is the same as  $C_{hole}$  in Equation (A7). We need to verify how many pairs of  $(a, b)$  exist in the inequation  $0 \leq -(a - M_2 + 1)\check{M}_1 - bM_2 \leq (\check{M}_1 - 1)(M_2 - 1)$  and  $M_1M_2 - 1 \leq a\check{M}_1 + (b + M_1 - 1)M_2 \leq \check{M}_1(M_2 - 1) + M_2(M_1 - 1)$ , where  $a \geq 0, b > 0$ . The above two inequations can be simplified to the same form as Equation (A8). Thus, the rest of the proof is completely the same as Proposition 1d.

### Appendix C. Proof of Proposition 3

(a) This property can be directly obtained from Proposition 1a and Proposition 2a.

(b) As  $\hat{\mathbb{L}}_{cd}^+ = \{\mathbf{M}_2\mathbf{m}_{cd} - \check{\mathbf{M}}_1\mathbf{n}_{cd} - \mathbf{v}_l\}$ ,  $\mathbf{n}_{cd} \in [-0.5(M_2 - 1) : 1 : 0.5(M_2 - 1)] \times [-(M_2 - 1) : 1 : 0]$ ,  $\mathbf{m}_{cd} \in [-0.5(M_1 - 1) : 1 : 0.5(M_1 - 1)] \times [0 : 1 : M_1 - 1]$  and  $\mathbf{v}_l = (0, -l)$ , we calculate the value range of  $y$  axis of  $\hat{\mathbb{L}}_{cd}^+$ . The result turns out to be

$$y \in [l, M_2(M_1 - 1) + \check{M}_1(M_2 - 1) + l]. \tag{A16}$$

Therefore, the mirrored  $\hat{\mathbb{L}}_{cd}^-$  has

$$y \in [-M_2(M_1 - 1) - \check{M}_1(M_2 - 1) - l, -l]. \tag{A17}$$

As  $l \geq 0$ ,  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cd}^-$  overlap on the  $y = 0$  axis if and only if  $l = 0$ . When  $l > 0$ , the value range of their  $y$  axis are completely apart.

(c) By analyzing the constuction process of CATSS, we can figure out that  $\hat{\mathbb{L}}_{cd}^+$  is derived from  $\tilde{\mathbb{L}}_{cd}^+$  with all its elements moving by vector  $\mathbf{u} = [(M_1 - 1)M_2 - (M_2 - 1)\check{M}_1]/2, -(M_2 - 1)\check{M}_1 - l$ . Thus, the consecutive range of  $\hat{\mathbb{L}}_{cd}^+$  can be expressed as

$$\begin{aligned} & \left\{ \hat{\mathbf{I}}_{cd} \mid \hat{\mathbf{I}}_{cd} \in [-(M_2 - 1), M_1M_2 - \check{M}_1(M_2 - 1) - 1]^2 - \mathbf{u} \right\} \\ & = \left\{ \hat{\mathbf{I}}_{cd} \mid \hat{\mathbf{I}}_{cd} \in [ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 ] \right. \\ & \quad \left. \times [(M_2 - 1)(\check{M}_1 - 1) + l : 1 : M_1M_2 + l - 1] \right\}. \end{aligned}$$

Similarly, the consecutive range of  $\hat{\mathbb{L}}_{cs}^+$  can be derived from Proposition 2b, i.e.,

$$\begin{aligned} & \left\{ \hat{\mathbf{I}}_{cs} \mid \hat{\mathbf{I}}_{cs} \in [(\check{M}_1 - 1)(M_2 - 1), M_1M_2 - 1]^2 + \mathbf{u} \right\} \\ & = \left\{ \hat{\mathbf{I}}_{cs} \mid \hat{\mathbf{I}}_{cs} \in [ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 ] \right. \\ & \quad \left. \times [-(M_2 - 1) - l : 1 : M_1M_2 - (M_2 - 1)\check{M}_1 - 1 - l] \right\}. \end{aligned}$$

(d) First of all, from Equations (33) and (35) we have

$$\begin{aligned} \widehat{\mathbb{L}}_{sd1} &= [-(M_2 - 1)\check{M}_1 : \check{M}_1 : (M_2 - 1)\check{M}_1]^2, \\ \widehat{\mathbb{L}}_{sd2} &= [-(M_1 - 1)M_2 : M_2 : (M_1 - 1)M_2]^2, \\ \widehat{\mathbb{L}}_{ss1}^+ &= [-(M_2 - 1)\check{M}_1 : \check{M}_1 : (M_2 - 1)\check{M}_1] \times [-2(M_2 - 1)\check{M}_1 - 2l : \check{M}_1 : -2l], \\ \widehat{\mathbb{L}}_{ss2}^+ &= [-(M_1 - 1)M_2 : M_2 : (M_1 - 1)M_2] \times [0 : M_2 : 2(M_1 - 1)M_2]. \end{aligned}$$

It is easy to find that  $\widehat{\mathbb{L}}_{sd2} \subseteq (\widehat{\mathbb{L}}_{ss2}^+ \cup \widehat{\mathbb{L}}_{ss2}^-)$ , and that when  $l = 0$ ,  $\widehat{\mathbb{L}}_{sd1} \subseteq (\widehat{\mathbb{L}}_{ss1}^+ \cup \widehat{\mathbb{L}}_{ss1}^-)$ . We can also confirm that the union of all self-coarrays  $\widehat{\mathbb{L}}_{self} = (\widehat{\mathbb{L}}_{sd} \cup \widehat{\mathbb{L}}_{ss}^+ \cup \widehat{\mathbb{L}}_{ss}^-)$  is still a sparse array in both  $x$  and  $y$  coordinates.

According to Equations (33) and (35), when either  $M_1$  or  $M_2$  is even, elements in  $\widehat{\mathbb{L}}_{cd}^+$ ,  $\widehat{\mathbb{L}}_{cd}^-$ ,  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$  fall in the positions whose  $x$  coordinates are  $0.5d$  away from the integers. However, from Equations (32) and (34) we know that elements in the self-coarrays fall in the integer positions. Thus,  $(\widehat{\mathbb{L}}_{sd} \cup \widehat{\mathbb{L}}_{ss}^+ \cup \widehat{\mathbb{L}}_{ss}^-) \cap (\widehat{\mathbb{L}}_{cd}^+ \cup \widehat{\mathbb{L}}_{cd}^- \cup \widehat{\mathbb{L}}_{cs}^+ \cup \widehat{\mathbb{L}}_{cs}^-) = \emptyset$ .

When both  $M_1$  and  $M_2$  are odd, elements in  $\widehat{\mathbb{L}}_{cd}^+$ ,  $\widehat{\mathbb{L}}_{cd}^-$ ,  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$  also fall in the integer positions. Note that  $\widehat{\mathbb{L}}_{self}$  has virtual sensors located at the integer positions around the origin with inter-element spacing  $\check{M}_1$  or  $M_2$ , and the boundary of  $\widehat{\mathbb{L}}_{sd2}$  and  $\widehat{\mathbb{L}}_{ss2}^+$  is larger than that of the consecutive part in Proposition 3c. There must be elements in  $\widehat{\mathbb{L}}_{self}$  falling in the range of the consecutive part in the cross-coarrays. Thus,  $(\widehat{\mathbb{L}}_{sd} \cup \widehat{\mathbb{L}}_{ss}^+ \cup \widehat{\mathbb{L}}_{ss}^-)$  is partly overlapped with  $(\widehat{\mathbb{L}}_{cd}^+ \cup \widehat{\mathbb{L}}_{cd}^- \cup \widehat{\mathbb{L}}_{cs}^+ \cup \widehat{\mathbb{L}}_{cs}^-)$ .

(e) Holes in  $\widehat{\mathbb{L}}_{cd}^+$  can be easily obtained by shifting the holes in CAACS (Proposition 1d) with  $-\mathbf{u}$ . Meanwhile, holes in  $\widehat{\mathbb{L}}_{cs}^+$  can also be easily obtained by shifting the holes in CAACS (Proposition 2d) with  $\mathbf{u}$ .

#### Appendix D. Proof of Proposition 4

(a) The first half of this property means that given an arbitrary integer point  $\widehat{\mathbf{l}}_{cdh} = (l_x, l_y)$  satisfying

$$l_y = -(a - M_2 + 1)\check{M}_1 - bM_2 + l, \tag{A18}$$

we intend to prove that there exist integer vectors  $\mathbf{m} = (x_1, y_1) \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n} = (x_2, y_2) \in [0 : 1 : M_2 - 1]^2$  such that  $\widehat{\mathbf{l}}_{cdh} = -\mathbf{M}_2\mathbf{m} - \check{\mathbf{M}}_1\mathbf{n} - \mathbf{v}_l$  holds. Actually, we only need to prove

$$-(a - M_2 + 1)\check{M}_1 - bM_2 + l = -M_2y_1 - \check{M}_1y_2 - [(M_2 - 1)\check{M}_1 - l] \tag{A19}$$

holds. By simplifying Equation (A19) we can get  $a\check{M}_1 + bM_2 = y_2\check{M}_1 + y_1M_2$ . Absolutely, we can find integer  $y_1$  and  $y_2$  to satisfy this relationship.

As for the second half of this property, we intend to prove that there exist integer vectors  $\mathbf{m} = (x_1, y_1) \in [0 : 1 : M_1 - 1]^2$  and  $\mathbf{n} = (x_2, y_2) \in [0 : 1 : M_2 - 1]^2$  such that  $\widehat{\mathbf{l}}_{csh} = \mathbf{M}_2\mathbf{m} - \check{\mathbf{M}}_1\mathbf{n} - \mathbf{v}_l$  holds, where  $\widehat{\mathbf{l}}_{csh} = (l_x, l_y)$  satisfies

$$l_y = a\check{M}_1 + bM_2 + l. \tag{A20}$$

We only need to prove

$$a\check{M}_1 + bM_2 + l = M_2y_1 - \check{M}_1y_2 + [(M_2 - 1)\check{M}_1 - l] \tag{A21}$$

holds. By simplifying Equation (A21) we can get  $bM_2 - (M_2 - a - 1)\check{M}_1 = y_1M_2 - y_2\check{M}_1$ . Since  $b > 0$  and  $M_2 - a - 1 < M_2$ , we can find integer  $y_1$  and  $y_2$  to satisfy this relationship.

(b) According to Proposition 3c, the upper bound of  $y$  coordinate in the consecutive range of  $\widehat{\mathbb{L}}_{cs}^+$  is  $M_1M_2 - (M_2 - 1)\check{M}_1 - 1 - l$ . Meanwhile, the lower bound of  $y$  coordinate in the consecutive range of  $\widehat{\mathbb{L}}_{cs}^-$  is  $-(M_1M_2 - (M_2 - 1)\check{M}_1 - 1) + l$ . Obviously, when  $l \in [0, M_1M_2 - (M_2 - 1)\check{M}_1 - 1]$ , the consecutive range of  $\widehat{\mathbb{L}}_{cs}^+$  and  $\widehat{\mathbb{L}}_{cs}^-$  are partly overlapped. When  $l$  grows to  $M_1M_2 - (M_2 - 1)\check{M}_1 - 1$ , only one overlapped row remains so that the hole-free URA reaches the largest range.

When  $l = M_1M_2 - (M_2 - 1)\check{M}_1$ , holes below  $\hat{\mathbb{L}}_{cs}^+$  are falling in the set  $\{(x, y) \mid y = (a - M_2 + 1)\check{M}_1 + (b + M_1 - 1)M_2 - l\} = \{(x, y) \mid y = a\check{M}_1 + (b - 1)M_2 - l\}$ . Since  $a \leq 0$  and  $b > 0$ , holes fall in the  $y = 0$  axis. For the mirrored  $\hat{\mathbb{L}}_{cs}^-$ , holes also fall in the  $y = 0$  axis. Thus, a broken row appears.

(c) This property can be easily obtained by combining the consecutive range of  $\hat{\mathbb{L}}_{cd}^+$  and  $\hat{\mathbb{L}}_{cs}^+$  in Proposition 3c. Since the four cross-coarrays complement their neighbors according to (a) and (b), the URA in the final diff-sum coarray ranges from the lower bound of  $\hat{\mathbb{L}}_{cd}^-$  to the upper bound of  $\hat{\mathbb{L}}_{cd}^+$ , i.e.,

$$\left\{ \hat{\mathbb{I}}_{DS} \mid \hat{\mathbb{I}}_{DS} \in \left[ -[(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 : 1 : [(M_1 - 1)M_2 - (M_2 - 1)(\check{M}_1 - 2)]/2 \right] \right. \\ \left. \times [-M_1M_2 - l + 1 : 1 : M_1M_2 + l - 1] \right\}.$$

## References

- Zhang, W.; Liu, W.; Wang, J.; Wu, S. Computationally efficient 2-D DOA estimation for uniform rectangular arrays. *Multidimens. Syst. Signal Process.* **2014**, *25*, 847–857. [\[CrossRef\]](#)
- Heidenreich, P.; Zoubir, A.M.; Rubsamen, M. Joint 2-D DOA estimation and phase calibration for uniform rectangular arrays. *IEEE Trans. Signal Process.* **2012**, *60*, 4683–4693. [\[CrossRef\]](#)
- Pal, P.; Vaidyanathan, P.P. Nested Arrays: A Novel Approach to Array Processing With Enhanced Degrees of Freedom. *IEEE Trans. Signal Process.* **2010**, *58*, 4167–4181. [\[CrossRef\]](#)
- Shen, Q.; Liu, W.; Cui, W.; Wu, S. Extension of nested arrays with the fourth-order difference co-array enhancement. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 2991–2995.
- Liu, J.; Zhang, Y.; Lu, Y.; Ren, S.; Cao, S. Augmented nested arrays with enhanced DOF and reduced mutual coupling. *IEEE Trans. Signal Process.* **2017**, *65*, 5549–5563. [\[CrossRef\]](#)
- Liu, C.L.; Vaidyanathan, P. Super nested arrays: Linear sparse arrays with reduced mutual coupling—Part I: Fundamentals. *IEEE Trans. Signal Process.* **2016**, *64*, 3997–4012. [\[CrossRef\]](#)
- Liu, C.L.; Vaidyanathan, P. Super nested arrays: Linear sparse arrays with reduced mutual coupling—Part II: High-order extensions. *IEEE Trans. Signal Process.* **2016**, *64*, 4203–4217. [\[CrossRef\]](#)
- Pal, P.; Vaidyanathan, P.P. Coprime sampling and the MUSIC algorithm. In Proceedings of the 2011 IEEE Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), Sedona, AZ, USA, 4–7 January 2011; pp. 289–294.
- Vaidyanathan, P.P.; Pal, P. Sparse Sensing With Co-Prime Samplers and Arrays. *IEEE Trans. Signal Process.* **2011**, *59*, 573–586. [\[CrossRef\]](#)
- Qin, S.; Zhang, Y.D.; Amin, M.G. Generalized Coprime Array Configurations for Direction-of-Arrival Estimation. *IEEE Trans. Signal Process.* **2015**, *63*, 1377–1390. [\[CrossRef\]](#)
- Adhikari, K.; Buck, J.R.; Wage, K.E. Extending coprime sensor arrays to achieve the peak side lobe height of a full uniform linear array. *EURASIP J. Adv. Signal Process.* **2014**, *2014*, 148. [\[CrossRef\]](#)
- Shen, Q.; Liu, W.; Cui, W.; Wu, S. Extension of co-prime arrays based on the fourth-order difference co-array concept. *IEEE Signal Process. Lett.* **2016**, *23*, 615–619. [\[CrossRef\]](#)
- Wang, W.; Ren, S.; Chen, Z. Unified coprime array with multi-period subarrays for direction-of-arrival estimation. *Digit. Signal Process.* **2018**, *74*, 30–42. [\[CrossRef\]](#)
- Liu, C.L.; Vaidyanathan, P. Remarks on the spatial smoothing step in coarray MUSIC. *IEEE Signal Process. Lett.* **2015**, *22*, 1438–1442. [\[CrossRef\]](#)
- Tayem, N.; Kwon, H.M. L-shape 2-dimensional arrival angle estimation with propagator method. *IEEE Trans. Antennas Propag.* **2005**, *53*, 1622–1630. [\[CrossRef\]](#)
- Niu, C.; Zhang, Y.; Guo, J. Interlaced double-precision 2-D angle estimation algorithm using L-shaped nested arrays. *IEEE Signal Process. Lett.* **2016**, *23*, 522–526. [\[CrossRef\]](#)
- Dong, Y.Y.; Dong, C.X.; Zhu, Y.T.; Zhao, G.Q.; Liu, S.Y. Two-dimensional DOA estimation for L-shaped array with nested subarrays without pair matching. *IET Signal Process.* **2017**, *10*, 1112–1117. [\[CrossRef\]](#)
- Hector, R.T.; Kassam, S.A. The unifying role of the coarray in aperture synthesis for coherent and incoherent imaging. *Proc. IEEE* **1990**, *78*, 735–752. [\[CrossRef\]](#)
- Greene, C.R.; Wood, R.C. Sparse array performance. *J. Acoust. Soc. Am.* **1978**, *63*, 1866–1872. [\[CrossRef\]](#)

20. Liu, C.L.; Vaidyanathan, P. Two-dimensional sparse arrays with hole-free coarray and reduced mutual coupling. In Proceedings of the 2016 50th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 6–9 November 2016; pp. 1508–1512.
21. Liu, C.L.; Vaidyanathan, P.P. Hourglass Arrays and Other Novel 2-D Sparse Arrays With Reduced Mutual Coupling. *IEEE Trans. Signal Process.* **2017**, *65*, 3369–3383. [[CrossRef](#)]
22. Ren, S.; Li, X.; Luo, X.; Wang, W. Extensions of Open Box Array with Reduced Mutual Coupling. *IEEE Sens. J.* **2018**, *18*, 5475–5484. [[CrossRef](#)]
23. Pal, P.; Vaidyanathan, P. Nested arrays in two dimensions, part I: Geometrical considerations. *IEEE Trans. Signal Process.* **2012**, *60*, 4694. [[CrossRef](#)]
24. Pal, P.; Vaidyanathan, P. Nested arrays in two dimensions, part II: Application in two dimensional array processing. *IEEE Trans. Signal Process.* **2012**, *60*, 4706–4718. [[CrossRef](#)]
25. Wu, Q.; Sun, F.; Lan, P.; Ding, G.; Zhang, X. Two-dimensional direction-of-arrival estimation for co-prime planar arrays: A partial spectral search approach. *IEEE Sens. J.* **2016**, *16*, 5660–5670. [[CrossRef](#)]
26. Zheng, W.; Zhang, X.; Zhai, H. Generalized coprime planar array geometry for 2-D DOA estimation. *IEEE Commun. Lett.* **2017**, *21*, 1075–1078. [[CrossRef](#)]
27. Zhang, X.; Zheng, W.; Chen, W.; Shi, Z. Two-dimensional DOA estimation for generalized coprime planar arrays: A fast-convergence trilinear decomposition approach. *Multidimens. Syst. Signal Process.* **2019**, *30*, 239–256 [[CrossRef](#)]
28. Vaidyanathan, P.; Pal, P. Sparse coprime sensing with multidimensional lattice arrays. In Proceedings of the 2011 Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), Sedona, AZ, USA, 4–7 January 2011; pp. 425–430.
29. Vaidyanathan, P.; Pal, P. Theory of sparse coprime sensing in multiple dimensions. *IEEE Trans. Signal Process.* **2011**, *59*, 3592–3608. [[CrossRef](#)]
30. Wang, X.; Chen, Z.; Ren, S.; Cao, S. DOA estimation based on the difference and sum coarray for coprime arrays. *Digit. Signal Process.* **2017**, *69*, 22–31. [[CrossRef](#)]
31. Chen, Z.; Ding, Y.; Ren, S.; Chen, Z. A Novel Nested Configuration Based on the Difference and Sum Co-Array Concept. *Sensors* **2018**, *18*, 2988. [[CrossRef](#)]
32. Shan, Z.; Yum, T.S. A conjugate augmented approach to direction-of-arrival estimation. *IEEE Trans. Signal Process.* **2005**, *53*, 4104–4109. [[CrossRef](#)]
33. Xu, G.; Kailath, T. Direction-of-arrival estimation via exploitation of cyclostationary—a combination of temporal and spatial processing. *IEEE Trans. Signal Process.* **1992**, *40*, 1775–1786. [[CrossRef](#)]
34. Mathews, C.P.; Zoltowski, M.D. Eigenstructure techniques for 2-D angle estimation with uniform circular arrays. *IEEE Trans. Signal Process.* **1994**, *42*, 2395–2407. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).