High-Order Sensor Array Geometries for Improved Direction of Arrival Estimation in Signal Processing

Author: Beining Zhou, St. Mark's School
Mentor: Hanshen Xiao

MIT PRIMES

July 2021

Abstract

In signal processing, the direction of arrival (DOA) estimation is a central problem to locate the source of a signal. It applies extensively in wireless communication systems such as radars and the GPS, in medical imaging, in telescopes, etc. Devising a signal sensor array geometry that achieves higher degree of freedom (DOF) has been a crucial challenge to improve the efficiency of DOA estimation. Recently, high-order cumulants are used extensively to construct high-order sensor arrays, but the state-of-the art high-order arrays are not optimal. This paper proposes novel sensor array geometries, the high-order embeded arrays (HOEA) for the 4th- and 6th-order and then extends those arrays to the $2q$th-order by layering. Compared to previous methods, the proposed HOEA significantly improves the DOF generation from $O(2^q N^{2q})$ to $O(17^{\frac{q}{3}} N^{2q})$, which increases the theoretical efficiency by 25% in the 4th order, 113% in the 6th, and 352% in the 12th order.

Keywords: direction of arrival estimation, sparse array, sparse ruler, high-order, difference co-array, nested array, coprime array

Contents

1 Introduction

1.1 Background

Signal processing is widely used throughout all aspects of our daily life to gather and process tremendous amounts of data. The direction of arrival (DOA) technique, which focuses on processing spatial estimation of the source of the signal, is applied in wireless communication systems such as radars and sonars, GPS systems and satellite navigation, medical imaging techniques, and radio telescopes.[1-5] Because of the extensive applications and the increasing amount of data to be gathered and processed, new methods need to be devised to process data faster at lower costs. This paper aims to achieve these goals by proposing a new sensor array geometries to increase the degree of freedom (DOF) in the DOA estimation.

The DOA estimation is a technique in sparse sensing, a signal processing method that fully recovers the spectral estimation of a signal using sparsely distributed sensors placed on a certain domain. Different from the traditional compress sensing, sparse sensing does not require the signal itself to be sparse. It comprises of two aspects: temporal estimation for signal frequency and spatial estimation for the DOA, or the angle in which the signal arrives.

DOA estimation is a foundational problem that is studied extensively in recent years. Prominent estimation methods include the widely known uniform linear array (ULA) which only renders DOF of linear efficiency. It is used to estimate the DOA with high-resolution subspace-based approaches such as the MUSIC and the ESPRIT have been proposed in [6] and [7]. However, these approaches were insufficient for under-determined DOA estimation, in which the number of sensors is less than the number of DOF required.

In recent years, two basic types of array geometries were proposed to address that problem. The coprime array and the nested array have significantly increased the DOF and have received wide research attention. Both of these structures are based on the sparse ruler problem and the difference co-array model, which will be introduced later in this paper. Variations of the nested array and the coprime array are proposed in [8] and [9] to reduce mutual coupling and improve efficiency.

Most recently, high-order cumulants, a statistical method, is used to generalize difference co-arrays into higher orders. However, generalizations from 2nd-order to high-order difference co-arrays are highly nontrivial. This is because any array with $2q$ number of sub-array would generate $\binom{2q}{q}$ number of permutation invariants, or sign combinations, in the high-order cumulants model. Most of the high-order arrays, including the state-of-the-art construction, sought to extend the nested array model. Notably, [10] proposed the idea of 4th-order cumulants, and [11] proposed the concept of the virtual 4th-order cumulant array. [12] and [13] proposed SAFOE-NA and SAFOE-CPA, which are extensions for nested arrays and coprime arrays into the 4th-order cumulant structure by using three subarrays. Improvements in their results include the EAS-NA-NA and EAS-NA-CPA in [14] and an extension of the coprime array in [15]. Various efforts have further generalized these structures to the arbitrary even-order cumulant for the difference co-array. The concept of $2q$-th order cumulants is proposed in [16]. Multi-order nested arrays such as the ML-NA and the SE-ML-NA have been further proposed in [17] and [18] to provide systematic ways to extend the results. Although these approaches extend the array geometry to $2q$-th order, they do not take full advantage of all the permutation invariants, which makes those structures not optimal.

1.2 Overview of the Proposed HOEA

This paper will explore a new model of high-order difference co-arrays, the HOEA, with a novel approach of embedding two or three sets of subarrays inside a larger set of subarrays.

In section 2, this paper defines the high-order cumulants and high-order difference co-arrays. At the same time, it explains how high-order cumulants can be applied in the DOA estimation model. Based on the structure, in section 3, the two existing array structures are generalized. This generalization takes into account 2nd-order sum in addition to differences, and the effects of shifts in these arrays are explored. These generalizations will enable the exploitation of more permutation invariants in the proposed model.

Section 4 explains the proposed 4th-order and 6th-order arrays in detail. First, this section introduces the key ideas behind bringing the ideas behind nested array and coprime array together to form a larger, embedded structure. This model exploits variations in the permutation invariant of the cumulants to achieve higher DOF. The nested embedding efficiently generates 2nd-order lags, while the coprime configures the 2nd-order lags to match the sign invariant and shifts generated lags evenly in the co-domain. For instance, the 4th-order cumulant can be expressed as a two-order sum or difference cumulant of two difference co-array structures. However, the permutation invariant determines the signs and possible combinations and results in three sign combinations. While no previous method utilizes all three combinations, the proposed array accounts for all three permutation invariants by shifting the arrays so that the three combined cumulants cover an extended consecutive lag. Then, this section presents a similar yet more complex construction of this model in 6th-order difference co-arrays.

Section 5 extends the HOEA model to $2q$-th order difference co-arrays by introducing a layering method that takes in two different co-array structures and layers them with a nested model. This method results in a new co-array structure of a higher order and therefore generalizes the embedding structure to $2q$-th order for any q.

Section 6 provides a comparison among the proposed method and previous methods and explains limitations in the proposed model subjected to future improvements.

2 High-order Cumulants in DOA Estimation

In DOA Estimation, the signal is measured as complex waveforms. Consider a linear sensor array that contains N sensors, and suppose these sensors occupy positions $\mathbb{S}$, which can be expressed as

$$\mathbb{S} = \{p_1 \cdot d, p_2 \cdot d, \ldots, p_N \cdot d\} \tag{1}$$

The spacing parameter d satisfies $d \leq \frac{\lambda}{2}$, and λ is the smallest wavelength of the signal.

Suppose that there are L independent narrowband signals $\{\mathbf{s}_i(n) \mid i = 1, 2, \ldots, L\}$ whose respective DOA are $\theta_{[1:L]} = \{\theta_i \mid i = 1, 2, \ldots, L\}$. Then, the array model for the received signal sample is expressed by a vector,

$$\begin{aligned}\mathbf{x}(n) &= \sum_{i=1}^{L} \mathbf{a}(\theta_i)\mathbf{s}_i(n) + \mathbf{w}(n) \\ &= \mathbf{A}(\theta)\mathbf{s}(n) + \mathbf{w}(n)\end{aligned}$$

where $\mathbf{a}(\theta_i)$ is the steering vector. This vector represents the array responses due to recieving the signal at an angle θ. For instance, in figure 1, θ_l would result in a steering vector $\mathbf{a}(\theta_l)$

Figure 1: Direction of arrival and steering vectors

The steering vector can be expressed by:

$$\mathbf{a}(\theta_i) = [e^{-j\frac{2\pi p_1}{\lambda}\sin\theta_i}, e^{-j\frac{2\pi p_2}{\lambda}\sin\theta_i}, \ldots, e^{-j\frac{2\pi p_N}{\lambda}\sin\theta_i}]^T$$

$\mathbf{A}(\theta)$ is a $N \times L$ steering matrix with $\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \ldots, \mathbf{a}(\theta_L)]$, the signal vector $\mathbf{s}(n) = [\mathbf{s}_1(n), \mathbf{s}_2(n), \ldots, \mathbf{s}_L(n)]$, and $\mathbf{w}(n)$ is the noise vector which is independent from the signal.

Since there is a one-to-one relationship between the DOA of the received signal and the calculated steering vector, the DOA of a signal can be derived by observing it from a signal sensor array. The goal is, therefore, to estimate $\theta_{[1:L]}$

According to [16], the $2q$-th order circular cumulant matrix, $C_{2q,x}(l)$ can be calculated. It is given by

$$\begin{aligned}C_{2q,x}(k) = &\sum_{i=1}^{L} c_{2q,\mathbf{s}_i}[\mathbf{a}(\theta_i)^{\otimes k} \otimes \mathbf{a}(\theta_i)^{*\otimes q-k}] \\ &\times [\mathbf{a}(\theta_i)^{\otimes k} \otimes \mathbf{a}(\theta_i)^{*\otimes (q-k)}]^H + \sigma_{\mathbf{w}}^2 I_{N^q}\delta(q-1)\end{aligned}$$

Here, k serves as an index for the matrix arrangement with $0 \leq k \leq q$. c_{2q,s_i} is the $2q$-th order circular autocumulant of a particular signal s_i where

$$c_{2q,\mathbf{s}_i} = \mathrm{Cum}\left[\mathbf{s}_{i_1}(t), \cdots, \mathbf{s}_{i_q}(t), \mathbf{s}^*_{i_{q+1}}(t), \cdots, \mathbf{s}^*_{i_{2q}}(t)\right]$$

where with $\mathbf{s}_{i_j} = \mathbf{s}_i, 1 \leq j \leq 2q$. $\otimes$ represents the Kronecker product, $\{\cdot\}^*$ represents the conjugacy matrix, and $\{\cdot\}^H$ is the Hermitian transpose. $\{\cdot\}^{\otimes k}$ is defined as the $N^k \times 1$ matrix where

$$\mathbf{a}(\theta_i)^{\otimes k} = \underbrace{\mathbf{a}\left(\theta_i\right) \otimes \mathbf{a}\left(\theta_i\right) \otimes \cdots \otimes \mathbf{a}\left(\theta_i\right)}_{k \text{ times}}$$

$\sigma_{\mathbf{w}}^2 I_{N^q} \delta(q-1)$ gives the white Gaussian noise.

This cumulant matrix is related to the $2q$-th order difference co-array. For the vector $\mathbf{a}(\theta_i)^{\otimes k}$, its elements are expressed as

$$e^{j\frac{2\pi}{\lambda}\left(\sum_{i=1}^{l} p_{n_i} - \sum_{i=l+1}^{q} p_{n_i}\right)\sin\theta}, 1 \leq n_i \leq N$$

Each one of these elements correspond to a steering vector of a high-order difference co-array that can estimate a signal's DOA angle. However, these elements (DOF) are not necessarily distinct, and the distinct DOF generated depends on the geometry of the sensor array. If more distinct lags with limited sensor positions are generated, more efficient DOA estimation can be performed.

This observation reduces the problem to how much consecutive and distinct integer elements of $\sum_{i=1}^{l} p_{n_i} - \sum_{i=l+1}^{q} p_{n_i}$ a sensor position array can generate. Essentially, for a given array $\mathbb{S}$, the sum of any q elements can be obtained. This paper examines the difference between any two of these sums and seeks to generate as many consecutive integers as possible. This problem is described as lag generation, which is defined below.

Definition 2.1. *Consider a linear array $\mathbb{S}$ in the form of (1). The set of 2q-th order differences*

$$\mathbb{C}^{2q}(\mathbb{S}) = \Phi^{2q}(\mathbb{S}) \cdot d$$

is a set of 2q-th order lags where

$$\Phi^{2q}(\mathbb{S}) = \{\sum_{i=1}^{q} p_{n_i} - \sum_{i=q+1}^{2q} p_{n_i} \mid n_i \in [1, N]\}$$

Denote Φ^2 as Φ for short.

In addition, in further sections, this paper will construct array structures with embedding subarrays into the larger linear array. Therefore, define the following concept of lags with the predetermined signs of subarrays.

Definition 2.2. *Consider m number of linear subarrays of the larger array $\mathbb{S}$, where $\mathbb{S} = \mathbb{S}_1 \cup \mathbb{S}_2 \cup \cdots \cup \mathbb{S}_m$. Let*

$$\mathbb{S}_i' = \mathbb{S}_i \ or \ -\mathbb{S}_i$$

for $i = 1, 2, \ldots, m$. The set of m-th order lags with predetermined signs (expressed shortly as lags afterward in this paper) is given by

$$\Phi(\mathbb{S}_1', \mathbb{S}_2', \ldots, \mathbb{S}_m') = \{\sum_{i=1}^{m} p_{n_i} \mid n_i \in [1, N]\}$$

In this definition, when $m = 2q$ and when there are exactly q number of $\mathbb{S}_i$s where $\mathbb{S}'_i = \mathbb{S}_i$ and q number of $\mathbb{S}_i$s where $\mathbb{S}'_i = -\mathbb{S}_i$,

$$\Phi^{2q}(\mathbb{S}) \supset \Phi(\mathbb{S}'_1, \mathbb{S}'_2, \ldots, \mathbb{S}'_m)$$

The lags generated by the larger linear array $\mathbb{S}$ is made up of all possible combinations of lags with predetermined signs of its subarrays, and there are $\binom{2q}{q}$ of such combinations in total to produce a permutation of q positive elements and q negative elements. Therefore, these combination renders possibility to improve the DOF generation.

3 Generalization of Coprime and Nested Arrays

In the high-order cumulants mode, $2q$ subarrays are used to build up the larger linear array. Therefore, there are different combinations to assign the signs to each subarray generates different sets of lags. In order to manipulate those lags to generate consecutive differences, variations of arrays structures in the 2nd-order are analyzed. Results the existing two existing array structures, coprime arrays and nested arrays, are generalized to incorporate 2nd-order sum co-array in addition to that of differences. The effects of shifts in these two arrays are also explored. These generalizations to the original structure will enable the proposed techniques to exploit more permutation invariants in the model in future sections.

3.1 Shifted Nested Arrays

First, considered the sum co-array along with the difference co-array in the shifted nested array structure.

Definition 3.1. *Let $\mathbb{S} = \mathbb{A} \cup \mathbb{B}$ be a linear array. Then, it is a shifted nested array if it has the form:*

$$\mathbb{S}_1 = \{(n_1 N_2 + \delta_1)d \mid n_1 = 0, 1, 2, \ldots N_1\}$$
$$\mathbb{S}_2 = \{(n_2 + \delta_2)d \mid n_2 = 0, 1, 2, \ldots N_2\}$$

Say that $\mathbb{S}$ is shifted by factors of δ_1, δ_2.

The sensor positions based on the shifted nested array are shown in Figure 2. The red dots represent sensor positions in the first subarray, $\mathbb{S}_1$, and the blue dots represent those in the 2nd, $\mathbb{S}_2$.

Figure 2: Sensor positions in the shifted nested array

The following lemma proves several properties of the shifted nested array. These properties describe the relationship between the shifts, δ_1 and δ_2, applied to the two subarrays, and the consecutive lags that they generate.

Lemma 3.2. *Let $\mathbb{S} = \mathbb{S}_1 \cup \mathbb{S}_2$ be a shifted nested array shifted by factors of δ_1, δ_2. Then*

$$\begin{aligned}
\Phi(\mathbb{S}_1, -\mathbb{S}_2) &= \{\mu \mid -N_2 + \delta_1 - \delta_2 \leq \mu \leq N_1N_2 + \delta_1 - \delta_2\} \\
&\supset \{\mu \mid \delta_1 - \delta_2 \leq \mu \leq N_1N_2 + \delta_1 - \delta_2\} \\
\Phi(-\mathbb{S}_1, \mathbb{S}_2) &= \{\mu \mid -N_1N_2 - \delta_1 + \delta_2 \leq \mu \leq N_2 - \delta_1 + \delta_2\} \\
&\supset \{\mu \mid -N_1N_2 - \delta_1 + \delta_2 \leq \mu \leq -\delta_1 + \delta_2\} \\
\Phi(\mathbb{S}_1, \mathbb{S}_2) &= \{\mu \mid \delta_1 + \delta_2 \leq \mu \leq N_1N_2 + N_2 + \delta_1 + \delta_2\} \\
&\supset \{\mu \mid \delta_1 + \delta_2 \leq \mu \leq N_1N_2 + \delta_1 + \delta_2\}
\end{aligned}$$

Proof. Let $\mathbb{S}_1(n_1)$ and $\mathbb{S}_2(n_2)$ denote the n_1-th and n_2-th sensor's position of the subarray $\mathbb{S}_1$ and $\mathbb{S}_2$ respectively. Then,

$$\begin{aligned}
\mathbb{S}_1(n_1) - \mathbb{S}_2(n_2) &= (n_1N_2 + \delta_1)d - (n_2 + \delta_2)d \\
&= (n_1N_2 - n_2)d + (\delta_1 - \delta_2)d \\
-\mathbb{S}_1(N_1 - n_1) + \mathbb{S}_2(N_2 - n_2) &= -(N_1N_2 - n_1N_2 + \delta_1)d + (N_2 - n_2 + \delta_2)d \\
&= (n_1N_2 - n_2)d + (-N_1N_2 + N_2 - \delta_1 + \delta_2)d \\
\mathbb{S}_1(n_1) + \mathbb{S}_2(N_2 - n_2) &= (n_1N_2 + \delta_1)d + (N_2 - n_2 + \delta_2)d \\
&= (n_1N_2 - n_2)d + (\delta_1 + \delta_2 + N_2)d
\end{aligned}$$

The results from the lags of the original nested array in [8] proves this lemma. □

This lemma shows that the lags are shifted by the same factor as the subarrays are, with little disturbance. Those disturbances would be accounted for by selections of parameters as described in section 4. This property of the shifted nested array enables the model to add any factor of shifts to any of the $2q$ subarrays in the high-order models and calculate the exact lags that they produce.

3.2 Shifted Coprime Arrays

Shifting the coprime array enables us to explore relations among the embedded subarrays in the HOEA and utilize that relation to generate consecutive lags. This section will analyze the key property of the shifted coprime array that is later used in the proposed model.

Definition 3.3. *Let $\mathbb{S} = \mathbb{A} \cup \mathbb{B}$ be a linear array. Then, it is a shifted coprime array if it has the form:*

$$\begin{aligned}
\mathbb{S}_1 &= \{(n_1 + \delta_1)pd \mid n_1 = 0, 1, 2, \ldots N_1\} \\
\mathbb{S}_2 &= \{(n_2 + \delta_2)qd \mid n_2 = 0, 1, 2, \ldots N_2\}
\end{aligned}$$

Say that $\mathbb{S}$ is shifted by factors of δ_1, δ_2.

The sensor positions described by shifted coprime arrays are shown in Figure 3.

Figure 3: Sensor positions in the shifted coprime array

When considering the 2nd-order difference co-array of the coprime array, shifting both subarrays a certain factor related to p or q can result in a new array with the lags generated unchanged. The following lemma describes this idea.

Lemma 3.4. *Let $\mathbb{S}_1 \cup \mathbb{S}_2$, $\mathbb{S}_3 \cup \mathbb{S}_4$ be two coprime arrays shifted by integer distances δ_1, δ_2 and $\delta_1 - kq$, $\delta_2 - kp$ for some integer k. p, q satisfies that $\gcd(p,q) = 1$. They can be expressed by the following*

$$\begin{aligned}
\mathbb{S}_1 &= \{(n_1 + \delta_1)pd \mid n_1 = 0, 1, 2, \dots N_1\} \\
\mathbb{S}_2 &= \{(n_2 + \delta_2)qd \mid n_2 = 0, 1, 2, \dots N_2\} \\
\mathbb{S}_3 &= \{(n_1 + \delta_1 - kq)pd \mid n_1 = 0, 1, 2, \dots N_1\} \\
\mathbb{S}_4 &= \{(n_2 + \delta_2 - kp)qd \mid n_2 = 0, 1, 2, \dots N_2\}
\end{aligned}$$

Then,

$$\Phi(\mathbb{S}_1 \cup \mathbb{S}_2) = \Phi(\mathbb{S}_3 \cup \mathbb{S}_4)$$

Proof. Because of the symmetry between $\Phi(\mathbb{S}_1, -\mathbb{S}_2)$ and $\Phi(\mathbb{S}_2, -\mathbb{S}_1)$ (and respectively $\mathbb{S}_3$ and $\mathbb{S}_4$), considered one group without loss of generality.

$$\begin{aligned}
\mathbb{S}_3(n_1) - \mathbb{S}_4(n_2) &= (n_1 + \delta_1 - kq)pd - (n_2 + \delta_2 - kp)qd \\
&= (n_1 + \delta_1)pd - (n_2 + \delta_2)qd - kqpd + kpqd \\
&= (n_1 + \delta_1)pd - (n_2 + \delta_2)qd \\
&= \mathbb{S}_1(n_1) - \mathbb{S}_2(n_2)
\end{aligned}$$

$$\begin{aligned}
\Phi(\mathbb{S}_1, -\mathbb{S}_2) &= \{\mathbb{S}_1(n_1) - \mathbb{S}_2(n_2) \mid n_1 = 1, 2, \dots N_1, n_2 = 1, 2, \dots N_2)\} \\
&= \{\mathbb{S}_3(n_1) - \mathbb{S}_4(n_2) \mid n_1 = 1, 2, \dots N_1, n_2 = 1, 2, \dots N_2\} \\
&= \Phi(\mathbb{S}_3, -\mathbb{S}_4)
\end{aligned}$$

□

Because of this lemma, different "blocks" in a shifted coprime array can represent the same set of consecutive lags for the DOF if they differ by a certain factor of p or q. This concept is illustrated in the graph below. Figure 4 shows an example of the lags generated by the coprime numbers 3 and 4. The axis each shows the sensor positions, $-24, -21, \dots, -3, 0, 3, \dots, 33$ and $-24, -20, \dots, -4, 0, 4, \dots, 32$. Each integer in the block represents the lag generated by the two sensor positions on the axis.

Figure 4: DOF lag blocks in the shifted coprime array

This graph illustrates Lemma 3.4. Each color block represents a blocked set of generated lags, and the lags are repeated across the graph. The lags generated by certain sensors of a shifted coprime array can be shifted. This lemma allows us to calculate the number of consecutive lags that are generated in a shifted coprime array based on the blocks that sensor positions correspond to.

4 Lower-order Sensor Array Models

This section will build up the HOEA from the key ideas to the 4th-order arrays, then to the 6th-order.

4.1 The Key Structure

The previous methods for high-order difference co-arrays are mostly based on generalizing the nested array because the distribution of coprime numbers is harder to examine and manipulate in higher dimensions. However, previous methods have disadvantages. Most importantly, they cannot harness the extensive combinations of the high-order structure. Specifically, in a $2q$-th order co-array with $2q$ subarrays, there are $\binom{2q}{q}$ ways of permutations to assign a "+" or a "-" sign in front of each element in the array.

This section proposes a solution to this issue by embedding. For a 4th-order cumulant, for instance, 4 subarrays are taken. These 4 subarrays will be in two groups of 2, and each group will generate respective lags, which are not necessarily consecutive. There will be lags resulting from both the sum and the difference of the subarrays. These lags will come together under different permutation invariants to produce more sets of lags. The larger structure that unites the two groups will adopt a coprime structure, which will bring together different sets of lags to be consecutive. In this way, the nested subarrays can account for all the variations in the permutations, while the coprime arrays serve to incorporate those variations into the strict structure of high-order difference co-arrays to generate consecutive integer lags.

First, group the subarrays into two sets of q subarrays and denoted them as $\mathbb{P}$ and $\mathbb{Q}$. Nested arrays are used to generate sets of consecutive lags $\mathbb{L}_P$ and $\mathbb{L}_Q$ of q-th order co-arrays. These lags do not need to follow Definition 2.1. Instead, they are of the form

$$\{\sum_{i=1}^{k} p_{n_i} - \sum_{i=k+1}^{q} p_{n_i} \mid n_i \in [1, N], k \in \{0, 1, \dots q\}\} \in \mathbb{L}_P \text{ or } \mathbb{L}_Q$$

Since it does not take in exactly half elements with negative signs and half with positive signs, these lags account for all the possible sign permutations in the q-th order.

Then, $\mathbb{L}_P$ and $\mathbb{L}_Q$ are transformed to fit the coprime array structure. Multiply each element in each subarray of $\mathbb{P}$ by p and those in $\mathbb{Q}$ by q, with the greatest common divisor $\gcd(p, q) = 1$. For each ordered pair of subarrays $(\mathbb{P}_i, \mathbb{Q}_j)$ with $\mathbb{P}_i \in \mathbb{P}$ and $\mathbb{Q}_j \in \mathbb{Q}$, these two subarrays can be shifted by Lemma 2.4. Then, take multiple coprime arrays formed by $\mathbb{P}_i \cup \mathbb{Q}_j$. By Lemma 2.6, the coprime structures can be shifted so that $\mathbb{P}_i$ to the same lag interval while making $\mathbb{Q}_i$ different.

Last, the generated lags of the coprime arrays are adjusted so that they form a sequence of consecutive integer lags. To do that, consider the following lemma.

Lemma 4.1. *Let* $\mathbb{S}_1$ *and* $\mathbb{S}_2$ *be subarrays of a coprime array, where*

$$\mathbb{S}_1 = \{n_1 pd \mid n_2 = 0, 1, 2, \ldots, N_1\}$$
$$\mathbb{S}_2 = \{n_2 qd \mid n_1 = \left\lfloor \frac{-p}{2} \right\rfloor, \left\lfloor \frac{-p}{2} \right\rfloor + 1, \ldots, 0, \ldots, \left\lfloor \frac{p}{2} \right\rfloor\}$$

with $N_1 \geq q$. *Then, the lags generated covers*

$$\Phi(\mathbb{S}_1, \mathbb{S}_2) = \{\mu \mid -(N_1+1)p + \left\lfloor \frac{p}{2} \right\rfloor q \leq \mu \leq (N_1+1)p - \left\lfloor \frac{p}{2} \right\rfloor q\}$$

Proof. With out lost of generality, only consider all positive lags generated by

$$\Phi_+(\mathbb{S}_1, -\mathbb{S}_2) = \{\mid n_1 p - n_2 q \mid\}$$

where Φ_+ denotes the positive lags generated by the 2nd-order co-arrays. By the Bézout theorem, there exists a solution (x_0, y_0) to the equation

$$\mid x_0 p - y_0 q \mid = \mu$$

Then, this equation has the general solution

$$\begin{cases} x = x_0 + kq \\ y = y_0 + kp \end{cases} \quad \text{for } k \in \mathbb{Z}$$

Therefore, there exists a pair of solution (x_1, y_1) such that $y_1 \in [\left\lfloor \frac{-p}{2} \right\rfloor, \left\lfloor \frac{p}{2} \right\rfloor]$. Here, there is $x_1 \notin [N_1 + 1, \infty]$ since or else,

$$\begin{aligned} \mid x_1 p - y_1 q \mid &= x_1 p - y_1 q \\ &> (N_1+1)p - \left\lfloor \frac{-p}{2} \right\rfloor q \end{aligned}$$

Let $n_1 = x_1$ and $n_2 = y_1$ if $x_1 \geq 0$, and $n_1 = -x_1$ and $n_2 = -y_1$ otherwise. There is

$$\mid n_1 p - n_2 q \mid = \mu$$

for any $\mu \in \Phi_+(\mathbb{S}_1, \mathbb{S}_2)$. □

Therefore, by this theorem, if $\mathbb{P}_i$ are all shifted to $\{\left\lfloor \frac{-p}{2} \right\rfloor, \left\lfloor \frac{-p}{2} \right\rfloor + 1, \ldots, 0, \ldots, \left\lfloor \frac{p}{2} \right\rfloor\}$, and $\mathbb{Q}_i$ are spread out over the interval of $\{0, 1, 2, \ldots\}$, an efficient lag generation method can be obtained. Because of this property, the original shifts applied to the subarrays are carefully constructed to satisfy the condition of consecutiveness. The details for the construction is provided in the section below.

4.2 The 4th-Order Array

Construct a 4-order difference co-array. Choose integers p, q such that $\gcd(p, q) = 1$, $p \leq N_1 N_2$, $q \leq N_3 N_4$. Consider the set of sensors positions, $\mathbb{S} = \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{Q}_1 \cup \mathbb{Q}_2$ where

$$\begin{aligned}
\mathbb{P}_1 &= \{(n_1 N_2 + \left\lfloor \frac{q}{2} \right\rfloor) pd \mid n_1 = 0, 1, 2, \dots, N_1\} \\
\mathbb{P}_2 &= \{(n_2 + q) pd \mid n_2 = 0, 1, 2, \dots, N_2\} \\
\mathbb{Q}_1 &= \{(n_3 N_4 - \left\lfloor \frac{p}{2} \right\rfloor) qd \mid n_4 = 0, 1, 2, \dots, N_4\} \\
\mathbb{Q}_2 &= \{(n_4 - \left\lfloor \frac{p}{2} \right\rfloor) qd \mid n_5 = 0, 1, 2, \dots, N_5\}
\end{aligned}$$

Then, the 4-th order array will generate a set of consecutive lags

$$\Phi_C^4(\mathbb{S}) = \{\mu \mid -M_{max}^4 \leq \mu \leq M_{max}^4, \mu \in \mathbb{Z}\}$$

where the DOF is expressed by

$$\begin{aligned}
M_{max}^6 &= \begin{cases} \left\lfloor \frac{5}{2} pq \right\rfloor & \text{when q is even} \\ \\ \left\lfloor \frac{5}{2} pq \right\rfloor - q & \text{when q is odd} \end{cases} \\
&\leq \left\lfloor \frac{5}{2} N_1 N_2 N_3 N_4 \right\rfloor
\end{aligned}$$

This section will explain the design of the 4th-order array. However, a more detailed proof is provided in Appendix A.

First, the larger sensor array, $\mathbb{S}$, is split into four subarrays, $\mathbb{P}_1$, $\mathbb{P}_2$, $\mathbb{Q}_1$, and $\mathbb{Q}_2$. Consider $\mathbb{P}_1$ and $\mathbb{P}_2$ to be grouped and similarly $\mathbb{Q}_1$ and $\mathbb{Q}_2$. Each of these groups form a shifted nested array with that is multiplied by pd and qd respectively. Consider only positive lags that are generated in this case, because if two sensor positions $p_1 d - p_2 d = kd$ for some desired k, then taking $p_2 d - p_1 d = kd$.

Consider the following permutation invariants:

$$\begin{aligned}
&\pm (p_1 + p_2) - (q_1 + q_2) \\
&\pm (p_1 - p_2) - (q_1 + q_2) \\
&\pm (p_1 + p_2) + (q_1 + q_2)
\end{aligned}$$

where $p_1 \in \mathbb{P}_1$, $p_2 \in \mathbb{P}_2$, $q_1 \in \mathbb{Q}_1$, $q_2 \in \mathbb{Q}_2$. These 3 sets of permutations accounts for a total of 6 different cumulants. Here, the parenthesis grouped the elements of the cumulants into two different groups, which represents two embeded subarrays. By Lemma 3.2, the lags generated in each group can be calculated. These lags will form another 2nd-order shifted coprime structure since the subarrays are all multiples of pd and qd. By Lemma 3.4, these three sets of cumulants would generate three "blocks"of lags. The specifically designed shifts, $\delta(\mathbb{P}_1)$, $\delta(\mathbb{P}_2)$, $\delta(\mathbb{Q}_1)$, and $\delta(\mathbb{Q}_2)$ will account for producing consecutive lags, and by Lemma 4.1, these three shifted "blocks" would produce a DOF efficiency of $O(\frac{5}{2}n^4)$, where n represents the number of sensors in the subarrays. (See Appendix A for a detailed proof.)

The 4th-order array successfully uses all $\frac{\binom{4}{2}}{2} = 3$ permutations of the signs in the difference co-array. Overall, it yields the efficiency of $O(\frac{5}{2}n^4)$ for its consecutive lag generation.

4.3 The 6th-Order Array

The 6th-order array is constructed differently from the 4th-order array because it is more complicated to match the shifting process with the consecutive lags. This section presents the construction of a 6-order difference co-array. Choose integers p, q such that $\gcd(p, q) = 1$, $p \leq N_1N_2N_3$, $q < N_4N_5N_6$. Consider the set of sensors positions, $\mathbb{S} = \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{P}_3 \cup \mathbb{Q}_1 \cup \mathbb{Q}_2 \cup \mathbb{Q}_3$ where

$$
\begin{aligned}
\mathbb{P}_1 &= \{(n_1N_2N_3)pd \mid n_1 = 0, 1, 2, \ldots, N_1\} \\
\mathbb{P}_2 &= \{(n_2N_3 + q)pd \mid n_2 = 0, 1, 2, \ldots, N_2\} \\
\mathbb{P}_3 &= \{(n_3 + \left\lfloor \frac{3q}{2} \right\rfloor)pd \mid n_3 = 0, 1, 2, \ldots, N_3\} \\
\mathbb{Q}_1 &= \{(n_4N_5N_6 - \left\lfloor \frac{5p}{2} \right\rfloor)qd \mid n_4 = 0, 1, 2, \ldots, N_4\} \\
\mathbb{Q}_2 &= \{(n_5N_6 - \left\lfloor \frac{7p}{2} \right\rfloor)qd \mid n_5 = 0, 1, 2, \ldots, N_5\} \\
\mathbb{Q}_3 &= \{(n_6 - 5p)qd \mid n_6 = 0, 1, 2, \ldots, N_6\}
\end{aligned}
$$

Then, the set of consecutive lags

$$\Phi_C^6(\mathbb{S}) = \{\mu \mid -M_{max}^6 \leq \mu \leq M_{max}^6, \mu \in \mathbb{Z}\}$$

where

$$M_{max}^6 = \left\lfloor \frac{17}{2}pq \right\rfloor \leq \left\lfloor \frac{17}{2}N_1N_2N_3N_4N_5N_6 \right\rfloor$$

Similar to the 4th-order array, the 6th-order embedded array consists of 6 subarrays. However, these subarrays are grouped into two groups of 3rd-order arrays. Consider the combinations the following:

$$
\begin{array}{ccc}
-p_1 + (p_2 + p_3) & & -q_1 + (q_2 + q_3) \\
+p_1 - (p_2 - p_3) & \text{and} & +q_1 - (q_2 - q_3) \\
p_1 + (p_2 - p_3) & & +q_1 + (q_2 - q_3)
\end{array}
$$

There are 9 cumulants produced from this combination, where $p_1 \in \mathbb{P}_1$, $p_2 \in \mathbb{P}_2$,$p_3 \in \mathbb{P}_3$, $q_1 \in \mathbb{Q}_1$, $q_2 \in \mathbb{Q}_2$, $q_3 \in \mathbb{Q}_3$. Within each group of the embedded array, group the last two elements, as shown in the parenthesis, to generate the lags of the last two elements. Since $\mathbb{P}_1$ and $\mathbb{Q}_1$ are each multiplied by N_2N_3 and N_5N_6, the lags from the last two element forms another shifted array with the first element. Since the first group is multiplied by pd and the second by qd, by Lemma 3.2, Lemma 3.4, and Lemma 4.1, 9 consecutive "blocks" of lags is generated, producing an efficiency of $O(\frac{17}{2}n^6)$ numbers of DOF. (See Appendix B for a detailed proof.)

In general, the 6th order array reaches an efficiency of $\frac{17}{2}n^6$ for lag generation. It utilizes 9 out of $\frac{\binom{6}{3}}{2} = 10$ number of sign combinations.

5 $2q$-th Order Embedded Array with Layering

5.1 The Layering Technique

For any two array structures that generate consecutive integer lags, they can be manipulated to gain a higher-order difference co-array. This can be achieved by the following.

Theorem 5.1. *Suppose there are two difference co-arrays, which are of $2q_1$-th order and $2q_2$-th order respectively. They are of the form:*

$$\mathbb{S}_1 = \{\alpha_1 \cdot d, \alpha_2 \cdot d, \dots, \alpha_{N_1} \cdot d\}$$
$$\mathbb{S}_2 = \{\beta_1 \cdot d, \beta_2 \cdot d, \dots, \beta_{N_2} \cdot d\}$$

Suppose that the generate consecutive lags

$$\Phi^{2q_1}(\mathbb{S}_1) = \{-\mu_1 \le \mu \le \mu_1\} \tag{2}$$
$$\Phi^{2q_2}(\mathbb{S}_2) = \{-\mu_2 \le \mu \le \mu_2\} \tag{3}$$

Then, take the $2(q_1+q_2)$-th order difference co-array $\mathbb{S}_1 \cup \mathbb{S}_2'$ where

$$\mathbb{S}_2' = \{2\beta_1\mu_1 \cdot d, 2\beta_2\mu_1 \cdot d, \dots, 2\beta_{N_2}\mu_1 \cdot d\}$$

This difference co-array generates consecutive lags

$$\Phi^{2(q_1+q_2)}(\mathbb{S}_1 \cup \mathbb{S}_2') = \{-2\mu_1\mu_2 - \mu_1 \le \mu \le 2\mu_1\mu_2 + \mu_1\}$$

Proof. Without lost of generality, consider only positive lags. Consider any integer $\mu \in [-2\mu_1\mu_2 - \mu_1, 2\mu_1\mu_2 + \mu_1]$. There is the fact that μ can be uniquely expressed with

$$\mu = 2k \cdot \mu_1 + r$$

for some integers $k \in [0, \mu_2]$ and $r \in [-\mu_1 + 1, \mu_1]$. Since $\Phi^{2q_1}(\mathbb{S}_1)$ and $\Phi^{2q_2}(\mathbb{S}_2)$ covers integer lags expressed in (2) and (3) $r \in \Phi^{2q_1}(\mathbb{S}_1)$ and $k \in \Phi^{2q_2}(\mathbb{S}_2)$. Therefore, there exist

$$r = \sum_{i=1}^{q_1} \alpha_{n_i} - \sum_{i=q_1+1}^{2q_1} \alpha_{n_i}, n_i \in [1, N_1]$$
$$k = \sum_{i=1}^{q_2} \beta_{n_i} - \sum_{i=q_2+1}^{2q_2} \beta_{n_i}, n_i \in [1, N_2]$$

Then, there is

$$\begin{aligned}
\mu &= r + 2k \cdot \mu_1 \\
&= (\sum_{i=1}^{q_1} \alpha_{n_i} - \sum_{i=q_1+1}^{2q_1} \alpha_{n_i}) + 2 \cdot (\sum_{i=1}^{q_2} \beta_{n_i} - \sum_{i=q_2+1}^{2q_2} \beta_{n_i}) \cdot \mu_1 \\
&= (\sum_{i=1}^{q_1} \alpha_{n_i} - \sum_{i=q_1+1}^{2q_1} \alpha_{n_i}) + (\sum_{i=1}^{q_2} 2\beta_{n_i}\mu_1 - \sum_{i=q_2+1}^{2q_2} 2\beta_{n_i}\mu_1) \\
&\in \Phi^{2(q_1+q_2)}(\mathbb{S}_1 \cup \mathbb{S}_2')
\end{aligned}$$

□

5.2 Extending the HOEA to the $2q$-th order

Generalizations without layering are difficult to construct to construct. Out of the $\binom{2q}{2}$ number of possible sign combinations, only $2q$ can be guaranteed. This is because of the following: suppose there is a way to assign values to $\delta(\mathbb{P}_i), \delta(\mathbb{Q}_j), i = 1, 2, \ldots q, j = 1, 2, \ldots, q$ such that the $\binom{2q}{2}$ sign permutations each generate distinct consecutive lag intervals. Then, it yields $\binom{2q}{2}$ equations of the relationship between $\delta(\mathbb{P}_i), \delta(\mathbb{Q}_j)$ and the shifts of generated lags. Then, there is the matrix relationship

$$\mathcal{S}_\Phi \begin{bmatrix} \delta(\mathbb{P}) \\ \delta(\mathbb{Q}) \end{bmatrix} = \delta(\Phi^{2q})$$

Here, $\delta(\mathbb{P}) = [\delta(\mathbb{P}_1), \delta(\mathbb{P}_2), \ldots, \delta(PP_q)]^T$ and $\delta(\mathbb{Q}) = [\delta(\mathbb{Q}_1), \delta(\mathbb{Q}_2), \ldots, \delta(\mathbb{Q}_q)]^T$ are two $q \times 1$ matrices. $\mathcal{S}_\Phi$ is a $2q \times (\frac{\binom{2q}{2}}{2})$ sign matrix of all sign permutations, where each row contain q number of 1's and q number of -1's. $\delta(\Phi^{2q})$ contains the desired amount of shifting for the lags generated by each permutation. Since their are $2q$ pivots in this equation, there are only $2q$ permutations guaranteed to form consecutive lags.

Therefore, the layering serves as a more efficient way of generalizing the HOEA structure to the $2q$-th order. Applying the method proposed in Theorem 5.1, the $2q$-th order structure can be broken up into several 6th-order arrays. In the cases when $2q \equiv 2$ or $2q \equiv 4 \pmod 6$, a 4th-order array can then be layered onto an original nested array.

This method is efficient. In the case when $2q \equiv 0 \pmod 6$, the layered HOEA has the lag generation efficiency of

$$O(2 \cdot \frac{(2 \cdot \frac{17}{2} N^6)^{\frac{q}{3}}}{2}) = O(17^{\frac{q}{3}} N^{2q})$$

Similarly, when $2q \equiv 2$ or $2q \equiv 4 \pmod 6$, the layered HOEA has the lag generation efficiency of $O(2 \cdot 17^{\frac{q-1}{3}} N^{2q})$ and $O(5 \cdot 17^{\frac{q-2}{3}} N^{2q})$ respectively.

6 Conclusion and Future Direction

This paper explored the relationship between a generalized Golomb ruler and the high-order DOA estimation in sparse sensing. A new approach, the HOEA, has been proposed to study the high-order difference co-array structure and the generalized sparse ruler problem. It relies on embedding the shifted nested arrays into an extended and shifted coprime array structure. High-order difference co-arrays is then constructed using the 6th-order array and the layering technique presented in the paper, and a 4th-order array or a nested array is attached to the layered HOEA when necessary.

The $2q$-th order difference co-array structures still have room for improvement. This paper only shows the 4th- and 6th-order arrays because HOEAs are less efficient as explained in Section 4.2. Other structures can potentially use more combinations of the permutation invariant of the sign and therefore extend the DOF.

Yet, this novel approach results in an improved generation of consecutive integer lags, thus enhancing the generated DOF and makes the DOA estimation more efficient. The charts and graphs below illustrate this improvement by comparing to two prominent $2q$-th order difference co-array structures, the $2q$L-NA in [17] and the SE-$2q$L-NA in [18].

Array Structures	Number of Sensors	Consecutive Lag Generation Efficiency
$2q$L-NA	$\sum_{i=1}^{2q}(N_i - 1) + 1$	$O(2N^{2q})$
SE-$2q$L-NA	$\sum_{i=1}^{2q}(N_i - 1) + 1$	$O(2^q N^{2q})$
HOEA (proposed)	$\sum_{i=1}^{2q}(N_i + 1)$	$O(17^{\frac{q}{3}} N^{2q})$ when $q \equiv 0$ $O(2 \cdot 17^{\frac{q-1}{3}} N^{2q})$ when $q \equiv 1$ $O(5 \cdot 17^{\frac{q-2}{3}} N^{2q})$ when $q \equiv 2 \mod 3$

Table 1: Comparison of the Efficiency of Consecutive Lag Generation of the Layered HOEA with Previous $2q$-th Order Difference Co-array Structures

In comparison to the $2q$L-NA and the SE-$2q$L-NA, the HOEA is significantly more efficient at consecutive lag generation. To quantify this improvement in efficiency, consider the following comparison between theoretical DOF generated by the HOEA and the state-of-the-art SE-$2q$L-NA:

- In the 4th order, the HOEA performs 25% better than the SE-$2q$L-NA.
- In the 6th order, the HOEA performs 113% better than the SE-$2q$L-NA.
- In the 12th order, the HOEA performs 352% better than the SE-$2q$L-NA.
- And the performance advantage increases exponentially as q increases.

This superior performance translates to fewer sensor resources used and more accurate DOA estimation in signal processing.

7 Acknowledgements

I would like to thank Hanshen Xiao for mentoring me and supporting in this project. I would also like to thank the MIT PRIMES program for giving me the research opportunity and their support for this project.

A Proof for the 4th-Order Array

Construction. Choose integers p, q such that $\gcd(p,q) = 1$, $p \leq N_1N_2$, $q \leq N_3N_4$. Consider the set of sensors positions, $\mathbb{S} = \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{Q}_1 \cup \mathbb{Q}_2$ where

$$\begin{aligned}
\mathbb{P}_1 &= \{(n_1N_2 + \left\lfloor \frac{q}{2} \right\rfloor)pd \mid n_1 = 0,1,2,\ldots,N_1\} \\
\mathbb{P}_2 &= \{(n_2 + q)pd \mid n_2 = 0,1,2,\ldots,N_2\} \\
\mathbb{Q}_1 &= \{(n_3N_4 - \left\lfloor \frac{p}{2} \right\rfloor)qd \mid n_4 = 0,1,2,\ldots,N_4\} \\
\mathbb{Q}_2 &= \{(n_4 - \left\lfloor \frac{p}{2} \right\rfloor)qd \mid n_5 = 0,1,2,\ldots,N_5\}
\end{aligned}$$

Then, the set of consecutive lags

$$\Phi_C^4(\mathbb{S}) = \{\mu \mid -M_{max}^4 \leq \mu \leq M_{max}^4, \mu \in \mathbb{Z}\}$$

where

$$\begin{aligned}
M_{max}^6 &= \begin{cases} \left\lfloor \frac{5}{2}pq \right\rfloor & \text{when q is even} \\ \\ \left\lfloor \frac{5}{2}pq \right\rfloor - q & \text{when q is odd} \end{cases} \\
&\leq \left\lfloor \frac{5}{2}N_1N_2N_3N_4 \right\rfloor
\end{aligned}$$

Proof. Due to the symmetry of the high order difference co-arrays, consider only positive lags. First, consider the combinations of the sum and the difference sets of $\mathbb{P}_1$ and $\mathbb{P}_2$ and $\mathbb{Q}_1$ and $\mathbb{Q}_2$ as sets of shifted nested arrays. Applying Lemma 1, there is

$$\begin{aligned}
\Phi^2(\mathbb{P}_1, -\mathbb{P}_2) &= \{\mu p \mid \delta(\mathbb{P}_1, -\mathbb{P}_2) \leq \mu \leq N_1N_2 + \delta(\mathbb{P}_1, -\mathbb{P}_2), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{P}_1, \mathbb{P}_2) &= \{\mu p \mid \delta(\mathbb{P}_1, \mathbb{P}_2) \leq \mu \leq N_1N_2 + \delta(\mathbb{P}_1, \mathbb{P}_2), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{Q}_1, -\mathbb{Q}_2) &= \{\mu q \mid \delta(\mathbb{Q}_1, -\mathbb{Q}_2) \leq \mu \leq (N_3+1)N_4 + \delta(\mathbb{Q}_1, -\mathbb{Q}_2), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{Q}_1, \mathbb{Q}_2) &= \{\mu q \mid \delta(\mathbb{Q}_1, \mathbb{Q}_2) \leq \mu \leq (N_3+1)N_4 + \delta(\mathbb{Q}_1, \mathbb{Q}_2), \mu \in \mathbb{Z}\}
\end{aligned}$$

where

$$\begin{aligned}
\delta_1 = \delta(\mathbb{P}_1, -\mathbb{P}_2) &= \delta(\mathbb{P}_1) - \delta(\mathbb{P}_2) = \left\lfloor \frac{q}{2} \right\rfloor - q = \left\lfloor \frac{-q}{2} \right\rfloor \\
\delta_2 = \delta(\mathbb{P}_1, \mathbb{P}_2) &= \delta(\mathbb{P}_1) + \delta(\mathbb{P}_1) = \left\lfloor \frac{q}{2} \right\rfloor + q = \left\lfloor \frac{3q}{2} \right\rfloor \\
\delta_3 = \delta(\mathbb{Q}_1, -\mathbb{Q}_2) &= \delta(\mathbb{Q}_1) - \delta(\mathbb{Q}_2) = -\left\lfloor \frac{q}{2} \right\rfloor + \left\lfloor \frac{q}{2} \right\rfloor = 0 \\
\delta_4 = \delta(\mathbb{Q}_1, \mathbb{Q}_2) &= \delta(\mathbb{Q}_1) + \delta(\mathbb{Q}_2) = -\left\lfloor \frac{q}{2} \right\rfloor - \left\lfloor \frac{q}{2} \right\rfloor \\
&= -q \text{ or } -q+1 \text{ when q is even or odd}
\end{aligned}$$

Denote all the above generated with lags as shifted by a factor of δ_i $\Phi(\mathbb{S}_i)$ for $i = 1,2,3,4$ for the sake of notation simplicity.

Denote $\Phi_{CP}(\delta_a, \delta_b)$ as the lags of a coprime array generated by two subarrays, $\mathbb{S}_a$ and $\mathbb{S}_b$. Consider three group of coprime arrays, $\Phi_{CP}(\delta_1, \delta_3)$, $\Phi_{CP}(\delta_1, -\delta_3)$, and $\Phi_{CP}(\delta_2, \delta_4)$. The difference array of these arrays are 4-order difference co-arrays because the number of plus and minus signs in each group sum to 2 plus signs and 2 minus signs. Since the positions of all the sensors in $\mathbb{S}_1$ and $\mathbb{S}_2$ are multiples of p and the positions of all the sensors in $\mathbb{S}_3$ and $\mathbb{S}_4$ are multiples of q, these three groups of arrays from shifted coprime arrays.

The key approach is to shift δ_1 and δ_2 to $\lfloor \frac{-q}{2} \rfloor$ using Lemma 2. Then, if the lags from the δ_3 and δ_4 component cover an extended range, apply Lemma 3 to yield the result.

$$\begin{aligned}
\Phi(\mathbb{S}_1, -\mathbb{S}_3) &= \Phi_{CP}(\left\lfloor \frac{-q}{2} \right\rfloor, 0) \\
\Phi(\mathbb{S}_1, \mathbb{S}_3) &= \Phi(\mathbb{S}_1, -(-\mathbb{S}_3)) = \Phi_{CP}(\left\lfloor \frac{-q}{2} \right\rfloor, -1) \\
\Phi(\mathbb{S}_2, -\mathbb{S}_4) &= \Phi_{CP}(\left\lfloor \frac{3q}{2} \right\rfloor, -q \text{ or } -q+1) \\
&= \Phi_{CP}(\left\lfloor \frac{q}{2} \right\rfloor, -2q \text{ or } -2q+1)
\end{aligned}$$

Since $p \leq N_1N_2$ and $q \leq N_4N_5$, the 3 lags above can be spliced together to form an extended coprime array. The lags generate by all nine arrays is contained in the lags generated by $A \cup B$, where

$$\begin{aligned}
A &= \{n_1pd \mid n_1 = \left\lfloor \frac{-q}{2} \right\rfloor, \left\lfloor \frac{-q}{2} \right\rfloor + 1, \dots, \left\lfloor \frac{q}{2} \right\rfloor\} \\
B &= \{n_2qd \mid n_2 = -2q \text{ or } -2q+1, \dots, 0\}
\end{aligned}$$

By Lemma 3, Therefore, the number of total positive lags generated by $A \cup B$ is

$$\Phi(A \cup B) = \{\mu \mid 0 \leq \mu \leq \left\lfloor \frac{5pq}{2} \right\rfloor \text{ or } \left\lfloor \frac{5}{2}pq \right\rfloor - q\}$$

Therefore,

$$\Phi_C^4(\mathbb{S}) = \{\mu \mid -M_{max}^4 \leq \mu \leq M_{max}^4, \mu \in \mathbb{Z}\}$$

where

$$\begin{aligned}
M_{max}^6 &= \begin{cases} \lfloor \frac{5}{2}pq \rfloor \text{ when q is even} \\ \\ \lfloor \frac{5}{2}pq \rfloor - q \text{ when q is odd} \end{cases} \\
&\leq \left\lfloor \frac{5}{2} N_1N_2N_3N_4 \right\rfloor
\end{aligned}$$

The equality is reached if and only if $p = N_1N_2$, $q = N_4N_5$ and when q is even.

Proof. Because of the symmetry between $\Phi(\mathbb{S}_1, -\mathbb{S}_2)$ and $\Phi(\mathbb{S}_2, -\mathbb{S}_1)$ (and respectively $\mathbb{S}_3$ and $\mathbb{S}_4$), consider one group without loss of generality.

$$\begin{aligned}\mathbb{S}_3(n_1) - \mathbb{S}_4(n_2) &= (n_1 + \delta_1 - kq)pd - (n_2 + \delta_2 - kp)qd \\ &= (n_1 + \delta_1)pd - (n_2 + \delta_2)qd - kqpd + kpqd \\ &= (n_1 + \delta_1)pd - (n_2 + \delta_2)qd \\ &= \mathbb{S}_1(n_1) - \mathbb{S}_2(n_2)\end{aligned}$$

$$\begin{aligned}\Phi(\mathbb{S}_1, -\mathbb{S}_2) &= \{\mathbb{S}_1(n_1) - \mathbb{S}_2(n_2) \mid n_1 = 1, 2, \dots N_1, n_2 = 1, 2, \dots N_2)\} \\ &= \{\mathbb{S}_3(n_1) - \mathbb{S}_4(n_2) \mid n_1 = 1, 2, \dots N_1, n_2 = 1, 2, \dots N_2\} \\ &= \Phi(\mathbb{S}_3, -\mathbb{S}_4)\end{aligned}$$

This proves the thoerem. □

B Proof for the 6th-Order Array

Construction. Choose integers p, q such that $\gcd(p, q) = 1$, $p \le N_1N_2N_3$, $q < N_4N_5N_6$. Consider the set of sensors positions, $\mathbb{S} = \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{P}_3 \cup \mathbb{Q}_1 \cup \mathbb{Q}_2 \cup \mathbb{Q}_3$ where

$$\begin{aligned}\mathbb{P}_1 &= \{(n_1N_2N_3)pd \mid n_1 = 0, 1, 2, \dots, N_1\} \\ \mathbb{P}_2 &= \{(n_2N_3 + q)pd \mid n_2 = 0, 1, 2, \dots, N_2\} \\ \mathbb{P}_3 &= \{(n_3 + \left\lfloor \frac{3q}{2} \right\rfloor)pd \mid n_3 = 0, 1, 2, \dots, N_3\} \\ \mathbb{Q}_1 &= \{(n_4N_5N_6 - \left\lfloor \frac{5p}{2} \right\rfloor)qd \mid n_4 = 0, 1, 2, \dots, N_4\} \\ \mathbb{Q}_2 &= \{(n_5N_6 - \left\lfloor \frac{7p}{2} \right\rfloor)qd \mid n_5 = 0, 1, 2, \dots, N_5\} \\ \mathbb{Q}_3 &= \{(n_6 - 5p)qd \mid n_6 = 0, 1, 2, \dots, N_6\}\end{aligned}$$

Then, the set of consecutive lags

$$\Phi_C^6(\mathbb{S}) = \{\mu \mid -M_{max}^6 \le \mu \le M_{max}^6, \mu \in \mathbb{Z}\}$$

where

$$M_{max}^6 = \left\lfloor \frac{17}{2}pq \right\rfloor \le \left\lfloor \frac{17}{2}N_1N_2N_3N_4N_5N_6 \right\rfloor$$

Proof. Due to the symmetry of the high order difference co-arrays, consider only positive lags. First, consider the following combinations of the sum and the difference sets of $\mathbb{P}_2$ and $\mathbb{P}_3$ and $\mathbb{Q}_2$

and $\mathbb{Q}_3$ as sets of shifted nested arrays. Applying Lemma 1, there is

$$
\begin{aligned}
\Phi^2(\mathbb{P}_2, -\mathbb{P}_3) &= \{\mu p \mid \delta(\mathbb{P}_2, -\mathbb{P}_3) \le \mu \le N_2N_3 + \delta(\mathbb{P}_2, -\mathbb{P}_3), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{P}_2, \mathbb{P}_3) &= \{\mu p \mid \delta(\mathbb{P}_2, \mathbb{P}_3) \le \mu \le N_2N_3 + \delta(\mathbb{P}_2, \mathbb{P}_3), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{Q}_2, -\mathbb{Q}_3) &= \{\mu q \mid \delta(\mathbb{Q}_2, -\mathbb{Q}_3) \le \mu \le N_5N_6 + \delta(\mathbb{Q}_2, -\mathbb{Q}_3), \mu \in \mathbb{Z}\} \\
\Phi^2(\mathbb{Q}_2, \mathbb{Q}_3) &= \{\mu q \mid \delta(\mathbb{Q}_2, \mathbb{Q}_3) \le \mu \le N_5N_6 + \delta(\mathbb{Q}_2, \mathbb{Q}_3), \mu \in \mathbb{Z}\}
\end{aligned}
$$

where

$$
\begin{aligned}
\delta(\mathbb{P}_2, -\mathbb{P}_3) &= \delta(\mathbb{P}_2) - \delta(-\mathbb{P}_3) = q - \left\lfloor \frac{3q}{2} \right\rfloor = \left\lfloor \frac{-q}{2} \right\rfloor \\
\delta(\mathbb{P}_2, \mathbb{P}_3) &= \delta(\mathbb{P}_2) + \delta(\mathbb{P}_3) = q + \left\lfloor \frac{3q}{2} \right\rfloor = \left\lfloor \frac{5q}{2} \right\rfloor \\
\delta(\mathbb{Q}_2, -\mathbb{Q}_3) &= \delta(\mathbb{Q}_2) - \delta(-\mathbb{Q}_3) = -\left\lfloor \frac{7p}{2} \right\rfloor + 5p = \left\lfloor \frac{3p}{2} \right\rfloor \\
\delta(\mathbb{Q}_2, \mathbb{Q}_3) &= \delta(\mathbb{Q}_2) + \delta(\mathbb{Q}_3) = -\left\lfloor \frac{7p}{2} \right\rfloor - 5p = \left\lfloor \frac{-17p}{2} \right\rfloor
\end{aligned}
$$

Next, consider co-arrays of order 3 by treating $\mathbb{P}_2$ and $\mathbb{P}_3$ and $\mathbb{Q}_2$ and $\mathbb{Q}_3$ as entities that forms sets of nested arrays with $\mathbb{P}_1$ and $\mathbb{Q}_1$. Applying Lemma 1 again, there is

$$
\begin{aligned}
\Phi(-\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3) &= \{\mu p \mid \delta_1 \le \mu \le N_1N_2N_3 + \delta_1, \mu \in \mathbb{Z}\} \\
\Phi(\mathbb{P}_1, -\mathbb{P}_2, \mathbb{P}_3) &= \{\mu p \mid \delta_2 \le \mu \le N_1N_2N_3 + \delta_2, \mu \in \mathbb{Z}\} \\
\Phi(\mathbb{P}_1, \mathbb{P}_2, -\mathbb{P}_3) &= \{\mu p \mid \delta_3 \le \mu \le N_1N_2N_3 + \delta_3, \mu \in \mathbb{Z}\} \\
\Phi(-\mathbb{Q}_1, \mathbb{Q}_2, \mathbb{Q}_3) &= \{\mu q \mid \delta_4 \le \mu \le N_4N_5N_6 + \delta_4, \mu \in \mathbb{Z}\} \\
\Phi(\mathbb{Q}_1, -\mathbb{Q}_2, \mathbb{Q}_3) &= \{\mu q \mid \delta_5 \le \mu \le N_4N_5N_6 + \delta_5, \mu \in \mathbb{Z}\} \\
\Phi(\mathbb{Q}_1, \mathbb{Q}_2, -\mathbb{Q}_3) &= \{\mu q \mid \delta_6 \le \mu \le N_4N_5N_6 + \delta_6, \mu \in \mathbb{Z}\}
\end{aligned}
$$

where

$$
\begin{aligned}
&\delta_1 = -N_1N_2N_3 - \delta(\mathbb{P}_1) + \delta(\mathbb{P}_2, \mathbb{P}_3) = -N_1N_2N_3 - 0 + \left\lfloor \frac{5q}{2} \right\rfloor = -N_1N_2N_3 + \left\lfloor \frac{5q}{2} \right\rfloor \\
&\delta_2 = \delta(\mathbb{P}_1) - \delta(\mathbb{P}_2, -\mathbb{P}_3) = 0 - \left\lfloor \frac{-q}{2} \right\rfloor = \left\lceil \frac{q}{2} \right\rceil \\
&\delta_3 = \delta(\mathbb{P}_1) + \delta(\mathbb{P}_2, -\mathbb{P}_3) = 0 + \left\lfloor \frac{-q}{2} \right\rfloor = \left\lfloor \frac{-q}{2} \right\rfloor \\
&\delta_4 = -N_4N_5N_6 - \delta(\mathbb{Q}_1) + \delta(\mathbb{Q}_2 + \mathbb{Q}_3) = -N_4N_5N_6 + \left\lfloor \frac{5p}{2} \right\rfloor + \left\lfloor \frac{-17p}{2} \right\rfloor = -N_4N_5N_6 - 6p \\
&\delta_5 = \delta_{\mathbb{Q}_1} - \delta_{\mathbb{Q}_2 - \mathbb{Q}_3} = -\left\lfloor \frac{5p}{2} \right\rfloor - \left\lfloor \frac{3p}{2} \right\rfloor = -4p \\
&\delta_6 = \delta_{\mathbb{Q}_1} + \delta_{\mathbb{Q}_2 - Q_3} = \left\lfloor \frac{5p}{2} \right\rfloor + \left\lfloor \frac{3p}{2} \right\rfloor = -p
\end{aligned}
$$

Denote the above generated with lags as shifted by a factor of δ_i $\Phi(\mathbb{S}_i)$ for $i = 1, 2, 3, 4, 5, 6$ for the sake of notation simplicity. Consider two arrays, $\Phi(\mathbb{S}_i)$ and $\Phi(\mathbb{S}_j)$ for $i = 1, 2,$ or 3, $j = 4, 5,$

or 6, The difference array of $\Phi(\mathbb{S}_i)$ and $\Phi(\mathbb{S}_j)$ is a 6-order difference co-array because $\Phi(\mathbb{S}_i)$ and $\Phi(\mathbb{S}_j)$ each contains one element with a minus sign and two elements with plus signs. Since the positions of all the sensors in $\Phi(\mathbb{S}_i)$ are multiples of p and the positions of all the sensors in $\Phi(\mathbb{S}_j)$ are multiples of q, $\Phi(\mathbb{S}_i) \cup \Phi(\mathbb{S}_j)$ is a shifted coprime array.

Denote $\Phi_{CP}(\mathbb{A}, \mathbb{B})$ as the lags of a coprime array generated by two subarrays, $\mathbb{A}$ and $\mathbb{B}$ which are shifted by factors of δ_a and δ_b. The key approach is to shift δ_a to $\lfloor \frac{q}{2} \rfloor$ using Lemma 2. Then, if the lags from the δ_b component cover an extended range, apply Lemma 3 to yield the result.

$$
\begin{aligned}
\Phi(\mathbb{S}_1, -\mathbb{S}_4) &= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{5q}{2} \right\rfloor, -N_4N_5N_6 - 6p) \\
&= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{q}{2} \right\rfloor, -N_4N_5N_6 - 8p) \\
\Phi(\mathbb{S}_1, -\mathbb{S}_5) &= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{5q}{2} \right\rfloor, -4p) \\
&= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{q}{2} \right\rfloor, -6p) \\
\Phi(\mathbb{S}_1, -\mathbb{S}_6) &= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{5q}{2} \right\rfloor, -p) \\
&= \Phi_{CP}(-N_1N_2N_3 + \left\lfloor \frac{q}{2} \right\rfloor, -3p) \\
\Phi(\mathbb{S}_2, -\mathbb{S}_4) &= \Phi_{CP}(\left\lceil \frac{q}{2} \right\rceil, -N_4N_5N_6 - 6p) \\
&= \Phi_{CP}(\left\lceil \frac{-q}{2} \right\rceil, -N_4N_5N_6 - 7p) \\
\Phi(\mathbb{S}_2, -\mathbb{S}_5) &= \Phi_{CP}(\left\lceil \frac{q}{2} \right\rceil, -4p) = \Phi_{CP}(\left\lceil \frac{-q}{2} \right\rceil, -5p) \\
\Phi(\mathbb{S}_2, -\mathbb{S}_6) &= \Phi_{CP}(\left\lceil \frac{q}{2} \right\rceil, -p) = \Phi_{CP}(\left\lceil \frac{-q}{2} \right\rceil, -2p) \\
\Phi(\mathbb{S}_3, -\mathbb{S}_4) &= \Phi_{CP}(\left\lfloor \frac{-q}{2} \right\rfloor, -N_4N_5N_6 - 6p) \\
\Phi(\mathbb{S}_3, -\mathbb{S}_5) &= \Phi_{CP}(\left\lfloor \frac{-q}{2} \right\rfloor, -4p) \\
\Phi(\mathbb{S}_3, -\mathbb{S}_6) &= \Phi_{CP}(\left\lfloor \frac{-q}{2} \right\rfloor, -p)
\end{aligned}
$$

Since $p \leq N_1N_2N_3$ and $q \leq N_4N_5N_6$, the nine lags above can be spliced together to form an extended coprime array. The lags generate by all nine arrays is contained in the lags generated by $A \cup B$, where

$$
\begin{aligned}
A &= \{n_1pd \mid n_1 = \left\lfloor \frac{-q}{2} \right\rfloor, \left\lfloor \frac{-q}{2} \right\rfloor + 1, \ldots, \left\lfloor \frac{q}{2} \right\rfloor\} \\
B &= \{n_2qd \mid n_2 = -9p, -9p+1, \ldots, 0\}
\end{aligned}
$$

By Lemma 3, Therefore, the number of total positive lags generated by $A \cup B$ is

$$\Phi(A \cup B) = \{\mu \mid 0 \leq \mu \leq \left\lfloor \frac{17pq}{2} \right\rfloor\}$$

. Therefore,

$$\begin{aligned} M_{max}^{6} &= \left\lfloor \frac{17}{2} pq \right\rfloor \\ &\leq \left\lfloor \frac{17}{2} N_1 N_2 N_3 N_4 N_5 N_6 \right\rfloor \end{aligned}$$

The equality is reached if and only if $p = N_1 N_2 N_3$, $q = N_4 N_5 N_6$. $\square$

References

[1] N. A. Baig and A. Hussain, ”RADAR signal processing for target range, Doppler and DoA estimation,” *2017 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, Islamabad, 2017, pp. 820-825, doi: 10.1109/IBCAST.2017.7868151.

[2] Jianguo Huang, Lijie Zhang, Qunfei Zhang, Yong Jin and Min Jiang, ”Performance analysis of DOA estimation for MIMO sonar based on experiments,” *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, Cardiff, 2009, pp. 269-272, doi: 10.1109/SSP.2009.5278588.

[3] Moussa, Mohamed et al. “Direction of Arrival Estimation of GPS Narrowband Jammers Using High-Resolution Techniques.” *Sensors* (Basel, Switzerland) vol. 19,24 5532. 14 Dec. 2019, doi:10.3390/s19245532

[4] Bin Yang, Feng He, Jin Jin, Huagang Xiong, and Guanghan Xu. ”DOA estimation for attitude determination on communication satellites” *Chinese Journal of Aeronautics*, vol. 27, no. 3, 2014. doi:10.1016/j.cja.2014.04.010

[5] A. Vesa and A. Iozsa, ”Direction - of - Arrival estimation for uniform sensor arrays,” *2010 9th International Symposium on Electronics and Telecommunications*, Timisoara, 2010, pp. 249-252, doi: 10.1109/ISETC.2010.5679326.

[6] R. Schmidt, ”Multiple emitter location and signal parameter estimation,” in *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276-280, March 1986, doi: 10.1109/TAP.1986.1143830.

[7] R. Roy and T. Kailath, ”ESPRIT-estimation of signal parameters via rotational invariance techniques,” in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984-995, July 1989, doi: 10.1109/29.32276.

[8] J. Liu, Y. Zhang, Y. Lu, S. Ren and S. Cao, ”Augmented Nested Arrays With Enhanced DOF and Reduced Mutual Coupling,” in *IEEE Transactions on Signal Processing*, vol. 65, no. 21, pp. 5549-5563, 1 Nov.1, 2017, doi: 10.1109/TSP.2017.2736493.

[9] A. Raza, W. Liu and Q. Shen, ”Thinned Coprime Array for Second-Order Difference Co-Array Generation With Reduced Mutual Coupling,” in *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 2052-2065, 15 April15, 2019, doi: 10.1109/TSP.2019.2901380.

[10] J. -. Cardoso and E. Moulines, ”Asymptotic performance analysis of direction-finding algorithms based on 4th-order cumulants,” in *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 214-224, Jan. 1995, doi: 10.1109/78.365301.

[11] P. Chevalier and A. Ferreol, ”On the virtual array concept for the 4th-order direction finding problem,” in *IEEE Transactions on Signal Processing*, vol. 47, no. 9, pp. 2592-2595, Sept. 1999, doi: 10.1109/78.782217.

[12] Q. Shen, W. Liu, W. Cui and S. Wu, "Extension of nested arrays with the 4th-order difference co-array enhancement," 2016 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, 2016, pp. 2991-2995, doi: 10.1109/ICASSP.2016.7472226.

[13] Q. Shen, W. Liu, W. Cui and S. Wu, "Extension of Co-Prime Arrays Based on the Fourth-Order Difference Co-Array Concept," in *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 615-619, May 2016, doi: 10.1109/LSP.2016.2539324.

[14] J. Cai, W. Liu, R. Zong and Q. Shen, "An Expanding and Shift Scheme for Constructing Fourth-Order Difference Coarrays," in *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 480-484, April 2017, doi: 10.1109/LSP.2017.2664500.

[15] T. H. A. Mahmud, Z. Ye, K. Shabir, R. Zheng and M. S. Islam, "Off-Grid DOA Estimation Aiding Virtual Extension of Coprime Arrays Exploiting Fourth Order Difference Co-Array With Interpolation," in *IEEE Access*, vol. 6, pp. 46097-46109, 2018, doi: 10.1109/ACCESS.2018.2865419.

[16] P. Chevalier, A. Ferreol and L. Albera, "High-Resolution Direction Finding From Higher Order Statistics: The$2rmq$-MUSIC Algorithm," in *IEEE Transactions on Signal Processing*, vol. 54, no. 8, pp. 2986-2997, Aug. 2006, doi: 10.1109/TSP.2006.877661.

[17] P. Pal and P. P. Vaidyanathan, "Multiple Order Nested Array: An Efficient Geometry for $2q$th Order Cumulant Based Array Processing," in *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1253-1269, March 2012, doi: 10.1109/TSP.2011.2178410.

[18] Q. Shen, W. Liu, W. Cui, S. Wu and P. Pal, "Simplified and Enhanced Multiple Level Nested Arrays Exploiting High-Order Difference Co-Arrays," in *IEEE Transactions on Signal Processing*, vol. 67, no. 13, pp. 3502-3515, 1 July1, 2019, doi: 10.1109/TSP.2019.2914887.