



This is a repository copy of *An improved expanding and shift scheme for the construction of fourth-order difference co-arrays*.

White Rose Research Online URL for this paper:
<http://eprints.whiterose.ac.uk/134925/>

Version: Accepted Version

Article:

Cai, J., Liu, W. orcid.org/0000-0003-2968-2888, Zong, R. et al. (1 more author) (2018) An improved expanding and shift scheme for the construction of fourth-order difference co-arrays. *Signal Processing*, 153. pp. 95-100. ISSN 0165-1684

<https://doi.org/10.1016/j.sigpro.2018.06.025>

Reuse

This article is distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) licence. This licence only allows you to download this work and share it with others as long as you credit the authors, but you can't change the article in any way or use it commercially. More information and the full terms of the licence here: <https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

An Improved Expanding and Shift Scheme for the Construction of Fourth-Order Difference Co-Arrays

Jingjing Cai^{a,*}, Wei Liu^{b,**}, Ru Zong^a, Bin Wu^a

^a*Department of Electronic and Engineering, Xidian University, Xi'an, 710071, China*

^b*Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, S1 4ET, United Kingdom*

Abstract

An improved expanding and shift (IEAS) scheme for efficient fourth-order difference co-array construction is proposed. Similar to the previously proposed expanding and shift (EAS) scheme, it consists of two sparse sub-arrays, but one of them is modified and shifted according to a new rule. Examples are provided with the second sub-array being a two-level nested array (IEAS-NA), as such a choice can generate more fourth-order difference lags (FODLs), although with the same number of consecutive lags. Furthermore, the array aperture of IEAS-NA is always greater than the corresponding EAS structure, which helps improving the DOA estimation result. Simulations results are provided to show the improved performance by the proposed new scheme.

Keywords: Sparse arrays, fourth-order difference co-array, expanding and shift scheme, cumulant.

1. Introduction

Sparse arrays combined with the second-order difference co-array concept can provide much more degrees of freedom (DOFs) than traditional uniform linear arrays (ULAs), and many methods have been proposed for underdetermined direction of arrival (DOA) estimation based on such arrays, such as the spatial smoothing-based subspace methods [1, 2], or compressive

*Corresponding author: Tel.: +86-29-88204759; fax: +86-29-88204759.

**Corresponding author: Tel.: +44-114-2225813; fax: +44-114-2225834.

Email addresses: jjcai@mail.xidian.edu.cn (Jingjing Cai),
w.liu@sheffield.ac.uk (Wei Liu), zongru@xidian.edu.cn (Ru Zong),
bwu@xidian.edu.cn (Bin Wu)

sensing (CS)-based methods [3, 4, 5, 6, 7]. Two representative sparse array structures are the co-prime arrays (CPAs) [8, 3, 9] and the nested arrays (NAs) [10, 11, 12, 13, 14].

By exploiting additional statistical properties of the impinging signals, such as non-Gaussianity or non-stationarity, fourth-order difference co-arrays can be constructed for underdetermined DOA estimation with further increased DOFs [15, 16, 17, 18, 19, 20, 21, 22, 23]. Two of the sparse array construction methods specifically designed for fourth-order difference co-array generation are the extension based on CPAs and NAs in [23, 24], which we call the three sub-arrays (TSA) scheme in this paper, and the expanding and shift (EAS) scheme proposed in [25], which are based on modifying and combining two existing sparse array structures. It was shown that the EAS scheme can generate much more consecutive co-array virtual sensors than the TSA scheme.

In this work, we propose an improved version of the EAS scheme which can generate more DOFs with a larger aperture. Same as the EAS scheme, there are two sparse sub-arrays in the proposed improved EAS (IEAS) scheme: one of them is expanded and shifted, but the value of shift is concerned with the consecutiveness of the second sparse array. As an implementation of the EAS scheme, the EAS-NA structure, where the second sparse sub-array is based on the nested array, is the most efficient one, generating more consecutive fourth-order difference lags (FODLs) than the others [25]. Similarly, one implementation of the IEAS scheme is the IEAS-NA structure, where the second sparse sub-array is derived from a nested array, and is more efficient than the others. The number of consecutive FODLs for both EAS-NA and IEAS-NA is exactly the same, while the number of total FODLs of IEAS-NA is greater than that of EAS-NA. More importantly, the sensors of IEAS-NA are distributed in a wider range than EAS-NA, leading to a larger array aperture. These advantages result in a better DOA estimation performance compared to the original EAS-NA scheme.

This paper is organized as follows. The EAS scheme is reviewed in Sec. 2 and the proposed IEAS scheme is introduced in Sec. 3. Design examples and a comparison of different schemes are presented in Sec. 4. Simulation results are provided in Sec. 5 and conclusions are drawn in Sec. 6.

2. Review of The EAS Scheme

Assume a sparse array can be split into two separate sparse linear arrays (SLA), and the first sub-array has I sensors, while the second one has J

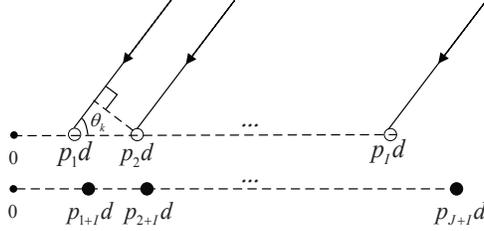


Figure 1: A sparse array structure consists of two sparse sub-arrays.

sensors as shown in Fig. 1. Denote the unit spacing by d , with $d = \lambda/2$. Then, positions of the whole array can be represented by a combination of the following two sets

$$\begin{aligned} P_1 &= \{p_1 \cdot d, p_2 \cdot d, \dots, p_I \cdot d\} \\ P_2 &= \{p_{1+I} \cdot d, p_{2+I} \cdot d, \dots, p_{J+I} \cdot d\} \end{aligned} \quad (1)$$

Suppose there are K far-field independent non-Gaussian narrowband signals $s_k(t)$ ($k = 1, \dots, K$) impinging on the array. With the DOA of the k th source being θ_k , and taking the origin “0” as the zero-phase reference point, the observed signal $x_i(t)$ at the i th sensor is given by

$$x_i(t) = \sum_{k=1}^K \exp(j2\pi p_i d \cos \theta_k / \lambda) s_k(t) + n_i(t) \quad (2)$$

$i = 1, \dots, I + J$, where $n_i(t)$ is the additive Gaussian noise of the i th sensor, independent of the signals. Suppose $1 \leq i, j, u, v \leq I + J$ and $\{i, j, u, v\} \in \mathbb{Z}$. The fourth-order cumulant value $C(i, -j, u, -v)$ of the i th, j th, u th and v th sensor observed signals can be expressed as [15]

$$\begin{aligned} C(i, -j, u, -v) &= \text{cum}[x_i(t), x_j^*(t), x_u(t), x_v^*(t)] \\ &= \sum_{k=1}^K \exp[j2\pi(p_i - p_j + p_u - p_v)d \cos \theta_k / \lambda] \cdot \\ &\quad \text{cum}(s_k(t), s_k^*(t), s_k(t), s_k^*(t)) \end{aligned} \quad (3)$$

where $()^*$ denotes complex conjugate, and $\text{cum}()$ denotes the fourth-order cumulant operation. The fourth-order difference co-array cannot only generate a much larger number of virtual sensors than the second-order difference co-array, but also remove the Gaussian noise components, further improving the DOA estimation performance of the whole system.

The FODL expression $(p_i - p_j + p_u - p_v)$ corresponding to the new virtual sensors can be written as

$$p_i - p_j + p_u - p_v = (p_i - p_j) + (p_u - p_v) \quad (4)$$

Clearly, this FODL expression can be seen as two second-order difference lags added together and we could construct the fourth-order difference co-array by two separate second-order difference co-arrays with different ranges as shown in the EAS scheme [25], where we first expand the spacing of the second sparse array, and then shift the expanded array by a proper distance.

For the EAS scheme, we assume two sparse arrays, which contains \bar{M} and \bar{N} sensors, separately. They can be represented as

$$\begin{aligned} \bar{Q}_1 &= \{\bar{q}_1 \cdot d, \bar{q}_2 \cdot d, \dots, \bar{q}_m \cdot d, \dots, \bar{q}_{\bar{M}} \cdot d\} \\ \bar{Q}_2 &= \{\bar{q}_{1+\bar{M}} \cdot d, \bar{q}_{2+\bar{M}} \cdot d, \dots, \bar{q}_{n+\bar{M}} \cdot d, \dots, \bar{q}_{\bar{N}+\bar{M}} \cdot d\}, \end{aligned} \quad (5)$$

and the number of consecutive second-order difference co-array sensors for these two sub-arrays are defined as $C_{\bar{M}}$ and $C_{\bar{N}}$, separately.

To construct P_1 and P_2 according to the EAS scheme using the elements of \bar{Q}_1 and \bar{Q}_2 , \bar{Q}_1 can be used as P_1 directly, but \bar{Q}_2 should be expanded and shifted to be P_2 . To be specific, the unit spacing of \bar{Q}_2 should be expanded to $C_{\bar{M}} \cdot d$ at first, and then shifted by a proper distance Δq , by which one sensor of the two sub-arrays will be co-located, so that one of the co-located sensors can be removed. To obtain a larger aperture, the last sensor of the first sub-array can coincide with the first sensor of the second sub-array. The number of physical sensors is then $I = \bar{M}$ for P_1 and $J = \bar{N} - 1$ for P_2 , and the elements of the EAS scheme can be defined as

$$\begin{aligned} p_i &= \bar{q}_m & i &= 1, \dots, I; m = 1, \dots, \bar{M} \\ p_j &= \bar{q}_{n+\bar{M}} \cdot C_{\bar{M}} + \Delta q & j &= 1, \dots, J; n = 2, \dots, \bar{N} \end{aligned} \quad (6)$$

where $\Delta q = \bar{q}_{\bar{M}} - \bar{q}_{1+\bar{M}} C_{\bar{M}}$, and $\bar{q}_{2+\bar{M}} \cdot C_{\bar{M}} + \Delta q$ is the first sensor of P_2 . Note that the first sensor $\bar{q}_{1+\bar{M}} \cdot C_{\bar{M}} + \Delta q$ of P_2 is co-located with $\bar{q}_{\bar{M}}$ and has been removed. As a result, the total number of sensors and FODLs of this scheme can be expressed as

$$\bar{L} = \bar{M} + \bar{N} - 1, \quad C_{\bar{L}} = C_{\bar{M}} C_{\bar{N}} \quad (7)$$

Define EAS-NA as the EAS scheme with the second sparse sub-array being a nested array. For a nested array, we have the property of $\bar{q}_{\bar{N}+\bar{M}} - \bar{q}_{1+\bar{M}} = \frac{C_{\bar{N}}-1}{2}$, where $\frac{C_{\bar{N}}-1}{2}$ is the maximum number of positive consecutive

second-order lags. For such a choice, an extra segment of $2(\bar{q}_M - \bar{q}_1)$ consecutive FODLs will be generated. Then, the EAS-NA structure can finally generate a number of consecutive FODLs as

$$C_{\bar{L}na} = C_{\bar{M}}C_{\bar{N}} + 2(\bar{q}_M - \bar{q}_1). \quad (8)$$

For the EAS-NA scheme, the number of total FODLs including consecutive and nonconsecutive ones, is the same as the number of consecutive FODLs, which is given by

$$C_{\bar{T}na} = C_{\bar{M}}C_{\bar{N}} + 2(\bar{q}_M - \bar{q}_1) \quad (9)$$

3. The Improved EAS (IEAS) Scheme

Compared to the EAS scheme, instead of shifting the second sub-array in such a way that the two sub-arrays have one pair of co-located sensors, the IEAS scheme shifts the second sub-array further away from the first one and there are no co-located sensors in the new structure. Define IEAS-NA as the class of IEAS schemes with the second sparse sub-array being a nested array. As shown later, this further shift will increase the total FODLs of the IEAS-NA. Moreover, the aperture of the IEAS-NA scheme is increased which helps improve the resolution of the whole system.

Assume there are two sparse arrays which contains M and N sensors, separately, as given below.

$$\begin{aligned} Q_1 &= \{q_1 \cdot d, q_2 \cdot d, \dots, q_m \cdot d, \dots, q_M \cdot d\} \\ Q_2 &= \{q_{1+M} \cdot d, q_{2+M} \cdot d, \dots, q_{n+M} \cdot d, \dots, q_{N+M} \cdot d\} \end{aligned} \quad (10)$$

Assume these two sub-arrays can generate C_M and C_N second-order difference lags, separately. The first step in IEAS is also expanding the second sparse array as in the EAS scheme, which can generate a consecutive integers segment from $-(C_M C_N - 1)/2$ to $(C_M C_N - 1)/2$. Assuming the shift distance now is denoted by $\Delta s \cdot d$, the elements of IEAS sub-arrays with $I = M$ and $J = N$ physical sensors can be represented by

$$\begin{aligned} p_i &= q_m & i &= 1, \dots, I; m = 1, \dots, M \\ p_j &= q_{n+M} \cdot C_M + \Delta s & j &= 1, \dots, J; n = 1, \dots, N \end{aligned} \quad (11)$$

Note that the FODLs generated by these two sub-arrays can be at least $C_M C_N$ by choosing p_i and p_j from P_1 while p_u and p_v from P_2 in the FODL expression $(p_i - p_j) + (p_u - p_v)$. Choosing a proper Δs could further

increase the number of FODLs, and we can also use other strategies, such as choosing p_i, p_j and p_v from P_1 while p_u from P_2 . With this choice, using $(p_i - p_j) + (-p_v)$ will generate consecutive integers by shifting the autocovariance of P_1 to every negative points of P_1 . The consecutive integers segment generated by the autocovariance of P_1 is from $-(C_M - 1)/2$ to $(C_M - 1)/2$ with 0 at its center. Then, the consecutive integers segment shifted to $-q_M$ will be from $-q_M - (C_M - 1)/2$ to $-q_M + (C_M - 1)/2$, while the segment shifted to $-q_1$ is from $-q_1 - (C_M - 1)/2$ to $-q_1 + (C_M - 1)/2$. Removing the overlapped integers between these two segments, the overall consecutive integers segment is from $-q_M - (C_M - 1)/2$ to $-q_1 + (C_M - 1)/2$ with 0 at its center. Then, the corresponding FODLs the combination $[(p_i - p_j) + (-p_v)] + p_u$ generates can be considered as the consecutive integers segment generated by $(p_i - p_j) + (-p_v)$ being shifted again with the elements of Q_2 at its center. Assuming from q_{U+M} to q_{V+M} is the maximum consecutive integer range in Q_2 , they can definitely generate consecutive FODLs from $(q_{U+M}C_M + \Delta s) - q_M - (C_M - 1)/2$ to $(q_{V+M}C_M + \Delta s) - q_1 + (C_M - 1)/2$. To maximize the segment of FODLs, we would like the first point of this consecutive FODLs segment to be next to the last point of the consecutive FODLs segment $C_M C_N$. In this case, the corresponding Δs can be calculated as

$$\begin{aligned}
q_{U+M}C_M + \Delta s - q_M - (C_M - 1)/2 &= (C_M C_N - 1)/2 + 1, \\
\rightarrow \\
\Delta s &= (C_M C_N - 1)/2 - q_{U+M}C_M + (C_M - 1)/2 + q_M + 1.
\end{aligned} \tag{12}$$

With this value of Δs , the maximum positive FODLs will be increased to $(V - U + 1)C_M + q_M - q_1 - 1$ and the overall consecutive FODLs will also increase accordingly. The total number of sensors and consecutive FODLs of this scheme can be expressed as

$$\begin{aligned}
L &= M + N \\
C_L &= C_M C_N + 2(V - U + 1)C_M + 2(q_M - q_1)
\end{aligned} \tag{13}$$

The physical layout of these two schemes is shown in Fig. 2, and the way of constructing the IEAS scheme is summarised in Tab. 1. The steps of constructing the EAS scheme is a little different, which only uses the steps 1, 2, 5, 6 and 7. The EAS scheme can remove one co-located sensor of the two sub-arrays by the shifting, while the IEAS scheme can generate more FODLs using the consecutiveness of the second sub-array.

It can be seen that the co-array extension ability of IEAS is related to the consecutiveness of the second sparse array. Among all the existing sparse arrays with the same number of physical sensors, the two-level nested array

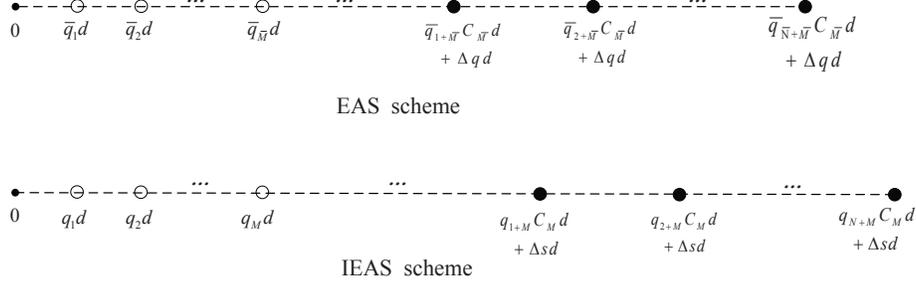


Figure 2: The physical layout of the EAS and IEAS schemes.

Table 1: Steps of the IEAS scheme

step 1	Calculate the second-order difference lags of Q_1 as C_M
step 2	Expand the inter-element of Q_2 to $C_M \cdot d$
step 3	Calculate the second-order difference lags of Q_2 as C_N
step 4	Count the consecutive sensors of Q_2 as $(V - U + 1)$
step 5	Obtain the last sensor position p_M of Q_1
step 6	Calculate the shift value of Δs
step 7	Shift Q_2 to Δs

contains the best consecutiveness in itself. It can be derived that the IEAS-NA can generate the most FODLs among all possible IEAS implementations based on existing sparse array structures. Let N_1 and N_2 be the two parameters of the second sparse array, and the maximum consecutive range of integers in the nested array is from 1 to $(N_1 + 1)$ with $(N_1 + 1)$ sensors in it, which means $V - U + 1 = N_1 + 1$, so that the number of consecutive FODLs is given by

$$C_{Lna} = C_M C_N + 2(N_1 + 1)C_M + 2(q_M - q_1) \quad (14)$$

Then, we can also derive the total number of FODLs of the IEAS-NA. For IEAS-NA, the consecutive FODLs correspond to the $N_1 + 1$ consecutive sensors of the second sparse array, and the total number of FODLs should also depend on the remaining $(N_2 - 1)$ nonconsecutive sensors of the second array. Assume in $(p_i - p_j) + (p_u - p_v)$, p_i, p_j and p_v are chosen from Q_1 while p_u is from Q_2 . The positive FODLs segments generated by choosing p_u from 1 to $N_1 + 1$ are consecutive, while segments generated by choosing p_u from $2(N_1 + 1)$ to $(N_2 - 1)(N_1 + 1)$ are nonconsecutive. Taking $p_u = 2(N_1 + 1)$ as an example, it generates a segment of FODLs from $2(N_1 + 1)C_M + \Delta s - q_1 - (C_M - 1)$ to $2(N_1 + 1)C_M + \Delta s - q_1 + (C_M - 1)/2$ with $(3C_M - 1)/2$ lags in

Table 2: Example settings and the FODLs generated by the EAS and IEAS schemes.

	EAS scheme	IEAS scheme
\bar{Q}_1 or Q_1	{1, 2, 3, 6, 9}	{1, 2, 3, 6, 9}
\bar{Q}_2 or Q_2	{1, 2, 3, 6}	{1, 2, 4}
P_1	{1, 2, 3, 6, 9}	{1, 2, 3, 6, 9}
P_2	{26, 43, 94}	{77, 94, 128}
	p_i, p_j from P_1, p_u, p_v from P_2	
$(p_i - p_j)$	-8 ~ 8	-8 ~ 8
$(p_u - p_v)$	-68, -51, -17, 0, 17, 51, 68	-51, -34, -17, 0, 17, 34, 51
$(p_i - p_j) + (p_u - p_v)$	-76 ~ -43, -25 ~ 25, 43 ~ 76	-59 ~ 59
	p_i, p_j, p_v from P_1, p_u from P_2	
$(p_i - p_j) + (-p_v)$	-17 ~ 7	-17 ~ 7
p_u	26, 43, 94	77, 94, 128
$(p_i - p_j) + (p_u - p_v)$	9 ~ 33, 26 ~ 50, 77 ~ 101	60 ~ 101, 111 ~ 135
	p_i, p_j, p_u from P_1, p_v from P_2	
$(p_i - p_j) + (p_u)$	-7 ~ 17	-7 ~ 17
$(-p_v)$	-94, -43, -26	-128, -94, -77
$(p_i - p_j) + (p_u - p_v)$	-101 ~ -77, -50 ~ -26, -33 ~ -9	-135 ~ -111, -101 ~ -60

it. Considering the negative lags, the total FODLs generated by the $N_2 - 1$ nonconsecutive sensors of the second array should be $(N_2 - 1)(3C_M - 1)$. That means the total FODLs of the IEAS-NA scheme is given by

$$C_{Tna} = C_M C_N + 2(N_1 + 1)C_M + 2(q_M - q_1) + (N_2 - 1)(3C_M - 1) \quad (15)$$

Now an example for both the EAS-NA and IEAS-NA schemes is provided, with the settings and generated FODLs listed in Tab. 2. The total number of sensors is $\bar{L} = L = 8$, and the shift value $\Delta s = 25$ for EAS-NA and $\Delta q = 60$ for IEAS-NA. It can be seen that the EAS-NA scheme generates $C_{\bar{L}} = 203$ and $C_{\bar{T}} = 203$ FODLs, while the IEAS-NA scheme generates $C_L = 203$ and $C_T = 253$ FODLs. The nonconsecutive FODLs of the IEAS scheme are in one positive segment from 111 to 135 and one negative segment from -135 to -111 with 25 lags in each of them.

4. Comparisons between IEAS-NA and EAS-NA

In this section, we give a comparison between IEAS-NA and EAS-NA, which are the most efficient schemes in IEAS and EAS, respectively. Given

the same number of physical sensors, we can have different sub-array parameters, which then results into different number of consecutive FODLs for the different schemes. To have a fair comparison, we choose the parameters giving the maximum number for each scheme, and compare the number of consecutive FODLs and total FODLs.

For IEAS-NA, the first sub-array contains M sensors while the second one contains N sensors; in EAS-NA, because the first sensor of the second sub-array is also the last sensor of the first sub-array, the number of sensors for the two sub-arrays is denoted by \bar{M} and $\bar{N} - 1$, separately. Let \bar{N}_1 and \bar{N}_2 denote the two parameters of the second sub-array in the EAS scheme. We have $\bar{N}_1 + \bar{N}_2 = \bar{N}$ and for IEAS-NA, $N_1 + N_2 = N$. Then, the total number of sensors can be written as

$$\bar{L} = \bar{M} + \bar{N}_1 + \bar{N}_2 - 1, \quad L = M + N_1 + N_2 \quad (16)$$

for EAS-NA and IEAS-NA, respectively; the number of consecutive FODLs $C_{\bar{L}}$ and C_L can be rewritten as

$$\begin{aligned} C_{\bar{L}na} &= C_{\bar{M}}(2\bar{N}_1\bar{N}_2 + 2\bar{N}_2 - 1) + 2(\bar{q}_{\bar{M}} - \bar{q}_1) \\ C_{Lna} &= C_M(2N_1N_2 + 2N_2 + 2N_1 + 1) + 2(q_M - q_1) \end{aligned} \quad (17)$$

The maximum value of $C_{\bar{L}}$ depends on the distribution of \bar{M} , \bar{N}_1 and \bar{N}_2 , while C_L depends on the distribution of M , N_1 and N_2 for C_L . Note that the structures of $C_{\bar{L}na}$ and C_{Lna} in (17) are almost the same. If we assume $\bar{L} = L$, $\bar{M} = M$ and $C_{\bar{M}} = C_M$, and with the assumption $N_1 + N_2 = \bar{N}_1 + \bar{N}_2 - 1 = N$, we next give a comparison between $C_{\bar{L}na}$ and C_{Lna} .

When $N_1 + N_2 = N$ is odd, $\bar{N}_1 + \bar{N}_2 = N + 1$ is even, vice versa. To get the maximum number of second-order difference lags for a nested array, the parameter settings should be

$$\begin{cases} \bar{N}_1 = \bar{N}_2 = (N + 1)/2 & N \text{ is odd} \\ \bar{N}_1 = N/2, \bar{N}_2 = N/2 + 1 & N \text{ is even} \end{cases} \quad (18)$$

$$\begin{cases} N_1 = (N - 1)/2, N_2 = (N + 1)/2 & N \text{ is odd} \\ N_1 = N_2 = N/2 & N \text{ is even} \end{cases}$$

Interestingly, with this setting the maximum value of $C_{\bar{L}na}$ and C_{Lna} will be exactly the same, given by

$$C_{Lmax} = C_{\bar{L}max} = \begin{cases} C_M[(N^2 + 1)/2 + 2N] + 2(q_M - q_1) & N \text{ is odd} \\ C_M(N^2/2 + 2N + 1) + 2(q_M - q_1) & N \text{ is even} \end{cases} \quad (19)$$

We can finally draw the conclusion that, when the total number of sensors for the EAS-NA and the IEAS-NA is the same, they can obtain the same number of consecutive FODLs.

In the same way, define $C_{\bar{T}max}$ and C_{Tmax} as the maximum of $C_{\bar{T}na}$ and C_{Tna} , separately. Because $C_{\bar{T}na}$ equals $C_{\bar{L}na}$, the settings of $C_{\bar{L}max}$ are suitable for $C_{\bar{T}max}$. While C_{Lna} is contained in C_{Tna} and also the main part of C_{Tna} , the settings of C_{Lmax} are also suitable for C_{Tmax} . These two maximum values can be given by

$$\begin{aligned}
C_{\bar{T}max} &= \begin{cases} C_M[(N^2 + 1)/2 + 2N] + 2(q_M - q_1) & N \text{ is odd} \\ C_M(N^2/2 + 2N + 1) + 2(q_M - q_1) & N \text{ is even} \end{cases} \\
C_{Tmax} &= \begin{cases} C_M[(N^2 + 1)/2 + 2N] + 2(q_M - q_1) & N \text{ is odd} \\ + (N - 1)(3C_M - 1)/2 & N \text{ is odd} \\ C_M(N^2/2 + 2N + 1) + 2(q_M - q_1) & N \text{ is even} \\ + (N - 2)(3C_M - 1)/2 & N \text{ is even} \end{cases}
\end{aligned} \tag{20}$$

Let the first sub-array of the IEAS-NA and EAS-NA schemes as the nested array (NA) and co-prime array (CPA), the resultant schemes are referred to as IEAS-NA-NA, EAS-NA-NA, IEAS-NA-CPA and EAS-NA-CPA, separately. We use M_1 and M_2 as the two parameters of the first sparse array for IEAS-NA, while \bar{M}_1 and \bar{M}_2 for EAS-NA. The parameter settings are listed in Tab. 3, which are chosen to give the maximum number of FODLs for each scheme, while the maximum number of consecutive FODLs and total FODLs are shown in Fig. 3.

Furthermore, the sensors of IEAS-NA are set in a wider range than EAS-NA. The last sensor location of these two schemes can also be derived as

$$\begin{aligned}
q_{\bar{L}} &= (\bar{N}_1\bar{N}_2 + \bar{N}_2 - 1)C_{\bar{M}} + \bar{q}_{\bar{M}} \\
q_L &= (2N_1N_2 + 2N_2 - 1)C_M + q_M
\end{aligned} \tag{21}$$

Their maximum values are

$$\begin{aligned}
q_{\bar{L}max} &= \begin{cases} C_M[(N^2 - 1)/4 + N] + q_M & N \text{ is odd} \\ C_M(N^2/4 + N) + q_M & N \text{ is even} \end{cases} \\
q_{Lmax} &= \begin{cases} C_M[(N^2 - 1)/2 + N] + q_M & N \text{ is odd} \\ C_M(N^2/2 + N - 1) + q_M & N \text{ is even} \end{cases}
\end{aligned} \tag{22}$$

Table 3: Parameter settings for different schemes.

L	EAS-NA-NA $(\bar{M}_1, \bar{M}_2,$ $\bar{N}_1, \bar{N}_2)$	IEAS-NA-NA $(M_1, M_2,$ $N_1, N_2)$	EAS-NA-CPA $(\bar{M}_1, \bar{M}_2,$ $\bar{N}_1, \bar{N}_2)$	IEAS-NA-CPA $(M_1, M_2,$ $N_1, N_2)$
4	(1, 1, 1, 2)	(1, 1, 1, 1)	--	--
5	(1, 2, 1, 2)	(1, 2, 1, 1)	--	--
6	(2, 2, 1, 2)	(2, 2, 1, 1)	--	--
7	(2, 3, 1, 2)	(2, 3, 1, 1)	--	--
8	(2, 3, 2, 2)	(2, 3, 1, 2)	(2, 3, 1, 2)	(2, 3, 1, 1)
9	(2, 3, 2, 3)	(2, 3, 2, 2)	(2, 3, 2, 2)	(2, 3, 1, 2)
10	(3, 3, 2, 3)	(3, 3, 2, 2)	(2, 3, 2, 3)	(2, 3, 2, 2)
11	(3, 4, 2, 3)	(3, 4, 2, 2)	(2, 3, 3, 3)	(2, 3, 2, 3)
12	(3, 4, 3, 3)	(3, 4, 2, 3)	(2, 3, 3, 4)	(2, 3, 3, 3)
13	(3, 4, 3, 4)	(3, 4, 3, 3)	(2, 3, 4, 4)	(2, 3, 3, 4)
14	(4, 4, 3, 4)	(4, 4, 3, 3)	(2, 3, 4, 5)	(2, 3, 4, 4)
15	(4, 5, 3, 4)	(4, 5, 3, 3)	(3, 4, 3, 4)	(3, 4, 3, 3)
16	(4, 5, 4, 4)	(4, 5, 3, 4)	(3, 4, 4, 4)	(3, 4, 3, 4)
17	(4, 5, 4, 5)	(4, 5, 4, 4)	(3, 4, 4, 5)	(3, 4, 4, 4)
18	(5, 5, 4, 5)	(5, 5, 4, 4)	(3, 4, 5, 5)	(3, 4, 4, 5)
19	(5, 6, 4, 5)	(5, 6, 4, 4)	(3, 4, 5, 6)	(3, 4, 5, 5)
20	(5, 6, 5, 5)	(5, 6, 4, 5)	(3, 4, 6, 6)	(3, 4, 5, 6)
21	(5, 6, 5, 6)	(5, 6, 5, 5)	(3, 4, 6, 7)	(3, 4, 6, 6)

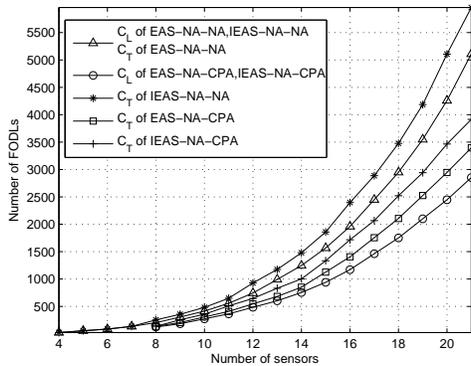


Figure 3: Number of FODLs with respect to the number of sensors.

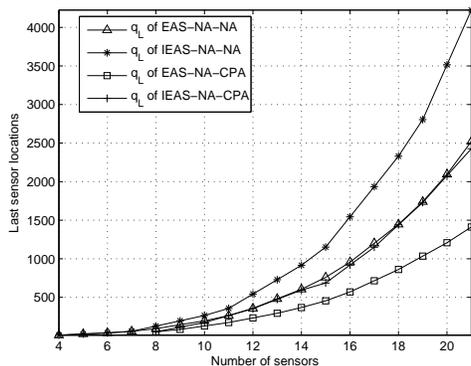


Figure 4: Last sensor position with respect to the number of sensors.

With the same setting as in Tab. 3, the values of $q_{\bar{L}}$ and q_L are shown in Fig. 4. From Figs. 3 and 4, we can see that the consecutive FODLs of IEAS-NA is always equal to that of EAS-NA, and the total FODLs of IEAS-NA is always greater than that of EAS-NA. Moreover, IEAS-NA always results in a larger physical aperture than EAS-NA.

5. Simulation Results

In this section, simulation results are provided to demonstrate the performance of the proposed IEAS-NA scheme. As proved, the IEAS-NA scheme always generates the same number of consecutive FODLs as the EAS-NA scheme, and the improvement by the IEAS-NA scheme lies in the increased total number of FODLs, which can not be exploited by the MUSIC-type al-

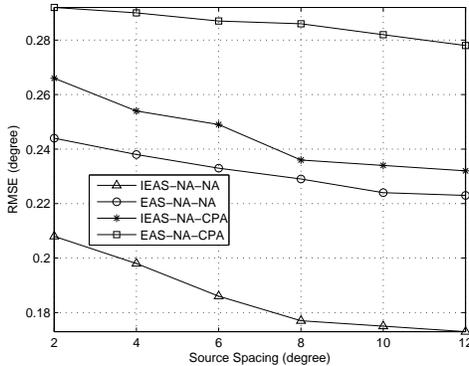


Figure 5: RMSE results with respect to source spacing.

gorithms, because these algorithms require a set of consecutive co-array lags due to the spatial smoothing operation adopted to deal with the coherent virtual sources arising from the co-array operation. On the other hand, the CS-based algorithm can use both consecutive and nonconsecutive lags of the co-array and therefore are employed here to show the better performance of the IEAS-NA algorithm.

In the CS-based DOA estimation algorithm, the constrained l_1 norm minimization problem can be solved using *cvx*, a package for specifying and solving convex problems [26, 27]. In the formulation, the full angle range from -90° to 90° is discretized with a step size of 0.05° . The total number of physical sensors is $L = 10$, and the parameter settings are according to Tab. 3. The resultant FODLs by EAS-NA-NA is $C_{\bar{L}} = C_{\bar{T}} = 413$, while it is $C_L = 413, C_T = 481$ for IEAS-NA-NA. Both EAS-NA-CPA and IEAS-NA-CPA can generate $C_{\bar{L}} = C_L = 273$ FODLs, while it is $C_{\bar{T}} = 273$ for EAS-NA-CPA and $C_T = 317$ for IEAS-NA-CPA.

In the first simulation, there are $K = 2$ narrowband source signals with different spacings. The input SNR is $0dB$, and the number of snapshots for calculating the fourth-order cumulant matrix is 20000. The DOA estimation result is shown in Fig. 5. Clearly, the root-mean-squared errors (RMSEs) of the IEAS-NA schemes is always lower than that of EAS-NA, due to a greater number of total FODLs and a larger aperture.

Then, we study the performance with different SNRs. The number of source signals is $K = 35$ and the number of snapshots for calculating the fourth-order cumulant matrix is 10000. The RMSE results obtained through 500 Monte Carlo trials are shown in Fig. 6 with a varied input SNR. Evi-

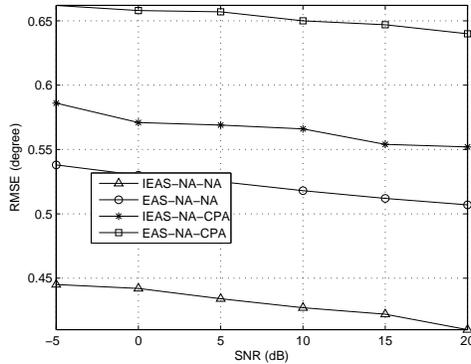


Figure 6: RMSE results with respect to input SNR.

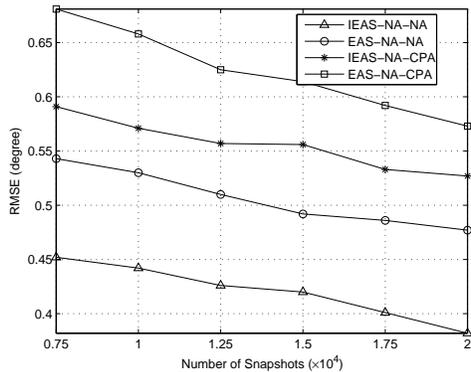


Figure 7: RMSE results with respect to the number of snapshots.

dently, the higher the input SNR, the higher its estimation accuracy. The performance of the IEAS-NA schemes is better than that of EAS-NA.

Next, we fix the input SNR to 0dB, and change the number of snapshots. The RMSE results are shown in Fig. 7, where we can see a similar trend and again the IEAS-NA-NA structure has provided the best result for the considered range of snapshot numbers.

At last, we use the 10-sensor IEAS-NA-NA array to give some idea about the number of required snapshots to provide a reasonable estimation result when the source number and SNR vary, and the results are shown in Figs. 8 and 9. The minimum number of snapshots is defined here as the minimum number needed to have 80% of Monte Carlo trials with an accuracy of less than or equal to 1° . The spacing between the sources is always 4° , and the

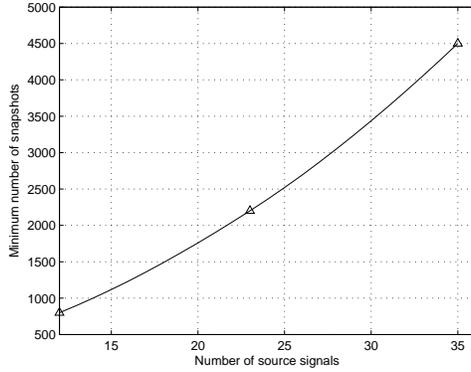


Figure 8: Minimum number of snapshots with respect to source number with a fixed SNR=0dB.

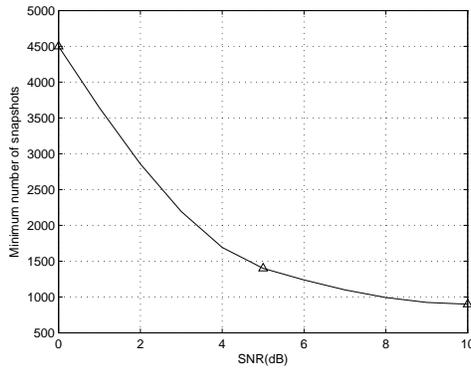


Figure 9: Minimum number of snapshots with respect to source SNR with 33 sources.

other settings are the same as before. As can be seen, at least about 500 to 1000 snapshots are needed to achieve a reasonable result.

6. Conclusion

A general sparse array construction scheme called improved expanding and shift (IEAS) has been proposed. It consists of two existing sparse sub-arrays, where the second sub-array is first expanded and then shifted to a proper position according to the consecutiveness of the second sub-array. The IEAS-NA scheme with the second sub-array being a two-level nested array is the most efficient one in the IEAS, for which more FODLs including consecutive and nonconsecutive ones are generated and its physical sensors are distributed in a wider physical range than the EAS-NA. As demonstrated

by simulation results, the proposed IEAS-NA scheme has achieved a much better performance than the existing EAS-NA.

7. Acknowledgement

This work was supported by the National Natural Science Foundation of China (61405150 and 61628101).

- [1] P. Pal and P. P. Vaidyanathan, "Coprime sampling and the MUSIC algorithm," in *Proc. IEEE Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE)*, Sedona, AZ, Jan. 2011, pp. 289–294.
- [2] C.-L. Liu and P. P. Vaidyanathan, "Remarks on the spatial smoothing step in coarray MUSIC," vol. 22, no. 9, pp. 1438–1442, Sep. 2015.
- [3] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 573–586, Feb. 2011.
- [4] Q. Shen, W. Liu, W. Cui, S. L. Wu, Y. D. Zhang, and M. Amin, "Low-complexity direction-of-arrival estimation based on wideband co-prime arrays," *IEEE Trans. Audio, Speech and Language Processing*, vol. 23, pp. 1445–1456, September 2015.
- [5] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Transactions on Signal Processing*, vol. 63, no. 6, pp. 1377–1390, March 2015.
- [6] J. J. Cai, D. Bao, and P. Li, "Doa estimation via sparse recovering from the smoothed covariance vector," *Journal of Systems Engineering and Electronics*, vol. 27, no. 3, pp. 555–561, June 2016.
- [7] Z. Shi, C. Zhou, Y. Gu, N. Goodman, and F. Qu, "Source estimation using coprime array: A sparse reconstruction perspective," *IEEE Sensors Journal*, vol. 17, no. 3, pp. 755–765, Feb. 2017.
- [8] P. Pal and P. P. Vaidyanathan, "Co-prime sampling and the music algorithm," in *IEEE Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE)*, Sedona, AZ, January 2011, pp. 289–294.

- [9] Y. M. Zhang, M. G. Amin, and B. Himed, “Sparsity-based DOA estimation using co-prime arrays,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 2013, pp. 3967–3971.
- [10] P. Pal and P. P. Vaidyanathan, “Nested arrays: a novel approach to array processing with enhanced degrees of freedom,” *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4167–4181, Aug. 2010.
- [11] Z. B. Shen, C. X. Dong, Y. Y. Dong, G. Q. Zhao, and L. Huang, “Broadband DOA estimation based on nested arrays,” *International Journal of Antennas and Propagation*, vol. 2015, 2015.
- [12] C. Liu and P. P. Vaidyanathan, “Super nested arrays: linear sparse arrays with reduced mutual coupling-part i:fundamentals,” *IEEE Transactions on Signal Processing*, vol. 64, no. 15, pp. 3997–4012, August 2016.
- [13] —, “Super nested arrays: linear sparse arrays with reduced mutual coupling-part ii:high-order extensions,” *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4203–4217, August 2016.
- [14] J. Liu, Y. Zhang, Y. Lu, S. Ren, and S. Cao, “Augmented nested arrays with enhanced dof and reduced mutual coupling,” *IEEE Transactions on Signal Processing*, vol. PP, no. 99, pp. 1–1, 2017.
- [15] J. F. Cardoso and E. Moulines, “Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants,” *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 214 –224, Jan. 1995.
- [16] M. C. Dogan and J. M. Mendel, “Applications of cumulants to array processing.i.aperture extension and array calibration,” *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1200 –1216, May 1995.
- [17] P. Chevalier and A. Ferreol, “On the virtual array concept for the fourth-order direction finding problem,” *IEEE Transactions on Signal Processing*, vol. 47, no. 9, pp. 2592 –2595, Sep. 1999.
- [18] P. Chevalier, L. Albera, A. Férréol, and P. Comon, “On the virtual array concept for higher order array processing,” *IEEE Transactions on Signal Processing*, vol. 53, no. 4, pp. 1254–1271, Apr. 2005.

- [19] P. Chevalier, A. Ferreol, and L. Albera, “High-resolution direction finding from higher order statistics: The 2q-music algorithm,” *IEEE Transactions on Signal Processing*, vol. 54, no. 8, pp. 2986–2997, Aug. 2006.
- [20] G. Birot, L. Albera, and P. Chevalier, “Sequential high-resolution direction finding from higher order statistics,” *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4144–4155, Aug 2010.
- [21] P. Pal and P. P. Vaidyanathan, “Multiple level nested array: an efficient geometry for 2qth order cumulant based array processing,” *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1253–1269, Mar. 2012.
- [22] W. K. Ma, T. H. Hsieh, and C. Y. Chi, “DOA estimation of quasi-stationary signals with less sensors than sources and unknown spatial noise covariance: a khatri-rao subspace approach,” *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2168–2180, April 2010.
- [23] Q. Shen, W. Liu, W. Cui, and S. L. Wu, “Extension of co-prime arrays based on the fourth-order difference co-array concept,” *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 615–619, May 2016.
- [24] ———, “Extension of nested arrays with the fourth-order difference co-array enhancement,” in *The 41st IEEE International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China, March 2016, pp. 2991–2995.
- [25] J. Cai, W. Liu, R. Zong, and Q. Shen, “An expanding and shift scheme for constructing fourth-order difference co-arrays,” *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 480–484, April 2017.
- [26] C. Research, “CVX: Matlab software for disciplined convex programming, version 2.0 beta,” <http://cvxr.com/cvx>, September 2012.
- [27] M. Grant and S. Boyd, “Graph implementations for nonsmooth convex programs,” in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, pp. 95–110, [http://stanford.edu/~boyd/graph/\\$_\\$.html](http://stanford.edu/~boyd/graph/$_$.html).