CO-PRIME ARRAY PROCESSING WITH SUM AND DIFFERENCE CO-ARRAY

Xiaomeng Wang[1], *Xin Wang*[1], *Xuehong Lin*[1,2]

[1] Department of Electrical and Computer Engineering, Stony Brook University, USA
[2] School of Information and Communication Engineering, Beijing University of Posts and Telecomm., China

ABSTRACT

Among various sparse-array techniques, co-prime array is found to be more attractive because of its high efficiency and simplicity. In this paper, we propose a scheme that can exploit both sum and difference co-arrays to form a larger size of continuous virtual array while keeping the conventional co-prime array configuration. Compared to the method that needs to extend the sensor array size to ensure the continuity of virtual array, our scheme can reduce the number of required physical sensors while still obtaining the same or even more degrees-of-freedom that can be exploited in direction-of-arrival (DOA) estimation. Simulation results demonstrate the effectiveness of our proposed method in achieving lower-cost, faster, and more accurate DOA estimation.

Index Terms— Co-prime array, direction-of-arrival (DOA) estimation, sum co-array, difference co-array.

1. INTRODUCTION

As one of the most significant topics in the signal processing field, array signal processing techniques have been studied extensively. One basic problem they can solve is to estimate the unknown parameters of the sources by exploiting the available temporal and spatial information collected by the sensor arrays. Specifically, they are often applied to estimate the direction-of-arrival (DOA) of sources, one major application of antenna array. Generally, a Uniform linear array (ULA) with $N+1$ elements can identify N sources, and has a degree of freedom (DOF) of N. To detect a large number of sources, it requires a large array with big N which would incur a big cost, and the estimation accuracy also reduces.

To address this challenge, sparse array construction such as minimum redundancy arrays (MRAs) [1], nested arrays [2] and co-prime arrays [3] have been proposed. These sparse arrays use their difference co-arrays to generate a virtual array with a larger size to increase DOF. Because there is no closed-form expression for the geometry configuration and no approximate DOF for MRAs, it is hard to design the MRA system in most cases. Co-prime array becomes more attractive because of its high efficiency and simplicity. With the use of co-prime array techniques, signal powers become the new sources. However, they are coherent, while the MUSIC algorithm [4] often used to estimate the DOA requires sources to be incoherent. Although spatial smoothing technique [5] can be used to decorrelate sources, it is generally applied on the uniform data. The difference co-array of a conventional co-prime array has "holes" [3], that is, some virtual array elements are missing. To solve this problem, Pal *et. al* in [6] consider using an extended co-prime construction, which requires $2M+N-1$ sensors to achieve $O(MN)$ degrees of freedom. The work in [7] uses proportional frequencies to fill holes in the virtual array. Besides the need of additional frequencies, these frequencies may not be available at the sources. Other works use compressive sensing [8] and temporal signal coherence (TCP) in moving co-prime arrays [9] to fill holes in the virtual array.

The aim of this work is to form a larger size continuous virtual array using only the configuration of the conventional co-prime array without the need of additional array elements or frequencies. Specifically, with a physical array of $M+N-1$ elements, we exploit the concurrent use of sum and difference co-arrays to form a continuous virtual array at low-cost. Compared to [6], this design can save M sensors and can be applied in systems where the available space and power are limited.

The remainder of the paper is organized as follows. In section 2, we review two configurations of co-prime array and their difference co-arrays. Section 3 introduces the sum co-array and the benefit of combining the sum and difference co-arrays. In Section 4, we propose a method to implement the proposed sum and difference co-array with physical array set to that of the conventional co-prime array. Several supporting simulation results are provided in Section 5 and conclusions are drawn in Section 6.

2. CO-PRIME ARRAYS AND CORRESPONDING DIFFERENCE CO-ARRAYS

In this section, we introduce two basic types of co-prime array configuration, conventional co-prime array and extended co-prime array, and their corresponding difference co-arrays.

2.1. Conventional Co-prime Arrays

A conventional co-prime array [3] consists of two uniform linear sub-arrays with separation Md and Nd respectively (Fig. 1). There are N sensors in the first sub-array and M sensors in the second sub-array. M and N are co-prime integers, i.e., $gcd(M,N)=1$, and d is the unit of inter-element spacing. To avoid spatial aliasing, d is typically set to $\lambda/2$, where λ is the wavelength of impinging narrowband signals. The sensors

This work was supported by the Office of Naval Research (ONR) under grant N00014-13-1-0209.

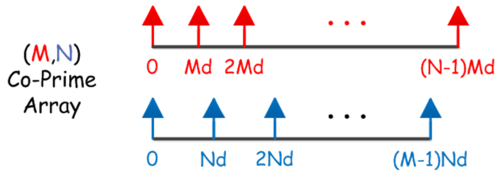


Fig. 1: Configuration of a conventional co-prime array

in the conventional co-prime array are positioned at

$$P_C = \{Mnd\} \cup \{Nmd\}, \tag{1}$$

where $0 \leq n \leq N-1$ and $0 \leq m \leq M-1$. Since the first sensors of these two uniform linear sub-arrays are co-located, the total number of sensors in the conventional co-prime array is $M+N-1$.

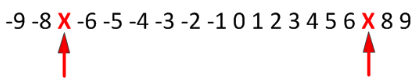


Fig. 2: The difference co-array of a conventional co-prime array

The corresponding difference co-array generated from this configuration can be expressed as

$$D_C = \{Mnd - Nmd\} \cup \{Nmd - Mnd\}, \tag{2}$$

where $0 \leq n \leq N-1$ and $0 \leq m \leq M-1$.

Fig. 2 shows an example of the difference co-array generated from a conventional co-prime array, with $M = 3$ and $N = 4$. We can see "holes" in the difference co-array, which prevents its direct application in many practical applications, including some DOA estimation cases that use the spatial smoothing technique [5] to de-correlate the coherent signals.

2.2. Extended Co-prime Arrays

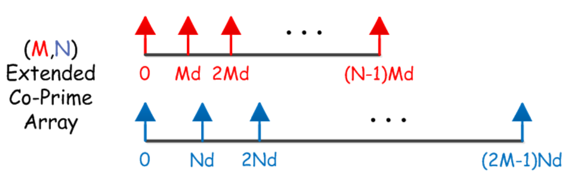


Fig. 3: Configuration of an extended co-prime array

To achieve a longer continuous virtual ULA from the difference co-array, extended co-prime array was proposed in [6]. As shown in Fig. 3, an extended co-prime array has similar configuration as that of the conventional co-prime array, but has more sensing elements in the second sub-array. The sensor positions in this modified configuration form the set

$$P_E = \{Mnd\} \cup \{Nmd\}, \tag{3}$$

where $0 \leq n \leq N-1$ and $0 \leq m \leq 2M-1$. The number of sensors in the second sub-array is doubled, so the total number of sensors in the extended co-prime array is $2M+N-1$.

Similarly, the corresponding difference co-array generated from the extended co-prime array can be expressed as

$$D_E = \{Mnd - Nmd\} \cup \{Nmd - Mnd\}, \tag{4}$$

where $0 \leq n \leq N-1$ and $0 \leq m \leq 2M-1$. Fig. 4 shows the difference co-array generated from an extended co-prime array with $M = 3$ and $N = 4$.

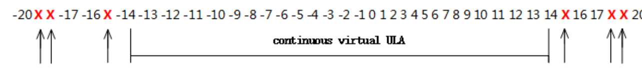


Fig. 4: The difference co-array of an extend co-prime array

The difference co-array generated from the extended co-prime array can form a virtual ULA whose continuous elements are at least located at

$$P_U = \{kd\}, \tag{5}$$

where $-MN \leq k \leq MN$. In this case, we can get $O(MN)$ DOFs using $2M+N-1$ physical sensors. Although the virtual array formulated is continuous, compared to the conventional co-array, it needs more sensors and increases the array aperture.

3. EXPLORATION OF SUM AND DIFFERENCE CO-ARRAYS

In this section, we first introduce the basic concept of the sum co-array and then show why we want to apply the sum co-array and difference co-array on the co-prime array simultaneously.

With the same configuration of the conventional co-prime array, i.e., formed with $M+N-1$ physical elements as in Fig. 1, the positive sum co-array can be expressed as

$$S_{C+} = \{Mnd + Nmd\} \cup \{2Mnd\} \cup \{2Nmd\}, \tag{6}$$

and the corresponding negative sum co-array generated from the conventional co-prime array can be expressed as

$$S_{C-} = \{-Mnd - Nmd\} \cup \{-2Mnd\} \cup \{-2Nmd\}, \tag{7}$$

where $0 \leq n \leq N-1$ and $0 \leq m \leq M-1$.

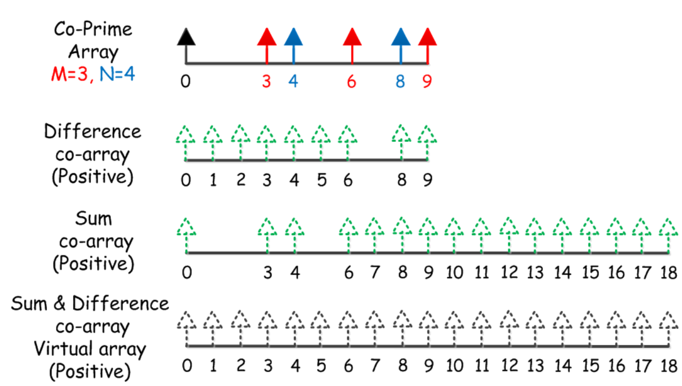


Fig. 5: Sum and difference co-array of a $M = 3$, $N = 4$ Co-Prime array

Fig. 5 shows the physical array structure of a conventional co-prime array with $M = 3$ and $N = 4$ and its positive sum and difference co-arrays. We can see that the sum co-array and the difference co-array have different holes. Therefore, we can exploit the sum and difference co-arrays together based on the same conventional co-prime array to form a longer continuous virtual ULA.

Lemma 1. *Given that M and N are co-prime integers, the set $\{\pm Mn \pm Nm\}$ at least contains integers from $-MN$ to MN.*

Proof. The Lemma is equivalent to: Given a k in the range $0 \leq k \leq MN$, we can always find n and m in the ranges $-(N-1) \leq n \leq N-1$ and $-(M-1) \leq m \leq M-1$, such that $k = Mn - Nm$. This follows from the Property 4 in [3]. □

Now theoretically, we can see that by generating both sum and difference co-arrays from a conventional (M, N) co-prime array, we can form a continuous virtual ULA from $-MN$ to MN so that we can obtain $O(MN)$ degrees of freedom using only $M + N - 1$ physical sensors. Compared with the extended co-prime array technique, this allows us to save M sensors. In practice, this continuous virtual ULA usually can be extended and range between $-MN-M-N+1$ and $MN+M+N-1$, which means we can obtain a DOF enhancement at the same time.

4. IMPLEMENTATION OF SUM AND DIFFERENCE CO-ARRAY

Our proposed Sum and Difference Co-Array only requires $M + N - 1$ physical sensors, the same as that of the conventional co-prime configuration. Although promising, there is a challenge of actually forming the continuous virtual array with the proposed array technique.

Usually, a difference co-array can be obtained by computing the covariance matrix of the observed data and the sum co-array arises naturally as the virtual array in active sensing [10]. However, this sum co-array contains only cross-sum terms and misses self-sum terms. Following we will present another approach to generate the sum and difference co-arrays for detecting the DOAs of a group of incoherent real-valued source signals.

Assuming D narrowband real-valued sources with powers $[\sigma_1^2\ \sigma_2^2\ \cdots\ \sigma_D^2]$ impinge on the array from directions $[\theta_1\ \theta_2\ \cdots\ \theta_D]$, the signals received at the array elements can be expressed as

$$\mathbf{x}[k] = \mathbf{A}\mathbf{s}[k] + \mathbf{n}[k] \tag{8}$$

where $\mathbf{A}$ is the array manifold matrix of the form

$$\mathbf{A} = [\mathbf{a}(\theta_1)\ \mathbf{a}(\theta_2)\ \cdots\ \mathbf{a}(\theta_D)]$$

$$= \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j(2\pi d/\lambda)M\sin\theta_1} & e^{j(2\pi d/\lambda)M\sin\theta_2} & \cdots & e^{j(2\pi d/\lambda)M\sin\theta_D} \\ \vdots & & & \\ e^{j(2\pi d/\lambda)(N-1)M\sin\theta_1} & e^{j(2\pi d/\lambda)(N-1)M\sin\theta_2} & \cdots & e^{j(2\pi d/\lambda)(N-1)M\sin\theta_D} \\ e^{j(2\pi d/\lambda)N\sin\theta_1} & e^{j(2\pi d/\lambda)N\sin\theta_2} & \cdots & e^{j(2\pi d/\lambda)N\sin\theta_D} \\ \vdots & & & \\ e^{j(2\pi d/\lambda)(M-1)N\sin\theta_1} & e^{j(2\pi d/\lambda)(M-1)N\sin\theta_2} & \cdots & e^{j(2\pi d/\lambda)(M-1)N\sin\theta_D} \end{pmatrix}$$

$\mathbf{s}[k] = [\mathbf{s}_1(k)\ \mathbf{s}_2(k)\ \cdots\ \mathbf{s}_D(k)]^T$ denotes the k_{th} snapshot of the source signal vector, and $\mathbf{n}[k]$ is the noise vector which is assumed to be temporally and spatially white and uncorrelated from the source.

In array signal processing, the difference co-array is formed naturally in the computation of the second order moments such as the autocorrelation between the received data,

$$\mathbf{R}_{xx1} = E[\mathbf{x}(k)\mathbf{x}(k)^H] = \mathbf{A}\mathbf{R}_{ss}\mathbf{A}^H + \sigma_D^2\mathbf{I}, \tag{9}$$

where $\mathbf{R}_{ss}$ is the source autocorrelation matrix, with

$$\mathbf{R}_{ss} = diag([\sigma_1^2\ \sigma_2^2\ \cdots\ \sigma_D^2]. \tag{10}$$

In practice, the autocorrelation matrix can be computed with the following sample average

$$\widehat{\mathbf{R}}_{xx1} = \frac{1}{L}\sum_{k=1}^{L}\mathbf{x}(k)\mathbf{x}(k)^H, \tag{11}$$

where L is the total number of snapshots. In order to build the new model using the difference co-array as the new array manifold matrix, we vectorize the autocorrelation matrix and get

$$\mathbf{z}_1 = vec(\mathbf{R}_{xx1}) = \mathbf{B}_1 \cdot \mathbf{p} + \sigma_n^2 vec(\mathbf{I}), \tag{12}$$

where $\mathbf{B}_1 = [\mathbf{B}_{\theta_1}\ \mathbf{B}_{\theta_2}\ \cdots \mathbf{B}_{\theta_D}] = \mathbf{A}^* \odot \mathbf{A}$ (Khatri-Rao product of $\mathbf{A}^*$ and $\mathbf{A}$) and $\mathbf{p} = [\sigma_1^2\ \sigma_2^2\ \cdots\ \sigma_D^2]$.

We consider the vector $\mathbf{z}_1$ to be the new received data, $\mathbf{B}_1$ to be the new array manifold matrix and $\mathbf{p}$ to be the new source signal. Similarly, we can apply autocorrelation-like computation to achieve the positive sum co-array $\mathbf{R}_{xx2}$ and the negative sum co-array $\mathbf{R}_{xx3}$ as

$$\mathbf{R}_{xx2} = E[\mathbf{x}(k)\mathbf{x}(k)^T] = \mathbf{A}\mathbf{s}\mathbf{s}^T\mathbf{A}^T + \mathbf{n}\mathbf{n}^T \tag{13}$$

$$\mathbf{z}_2 = vec(\mathbf{R}_{xx2}) = \mathbf{B}_2 \cdot \mathbf{p}_2 + vec(\mathbf{n}\mathbf{n}^T) \tag{14}$$

$$\mathbf{R}_{xx3} = E[\mathbf{x}(k)^*\mathbf{x}(k)^H] = \mathbf{A}^*\mathbf{s}^*\mathbf{s}^H\mathbf{A}^H + \mathbf{n}^*\mathbf{n}^H \tag{15}$$

$$\mathbf{z}_3 = vec(\mathbf{R}_{xx3}) = \mathbf{B}_3 \cdot \mathbf{p}_3 + vec(\mathbf{n}^*\mathbf{n}^H) \tag{16}$$

Since the source signals are real-valued,

$$\mathbf{p}_2 = \mathbf{p}_3 = \mathbf{p} = [\sigma_1^2\ \sigma_2^2\ \cdots\ \sigma_D^2]^T \tag{17}$$

We can then easily integrate the three newly generated received data vectors:

$$\mathbf{z} = [\mathbf{z}_1^T\ \mathbf{z}_2^T\ \mathbf{z}_3^T]^T \tag{18}$$

Thus, we get a larger corresponding array manifold matrix:

$$\mathbf{B} = [\mathbf{B}_1^T\ \mathbf{B}_2^T\ \mathbf{B}_3^T]^T \tag{19}$$

As there exist redundant and out-of-order elements in the vector, we have to drop and reorder some elements to rebuild $\mathbf{z}$ to form a new vector $\mathbf{z}'$ so that its corresponding $\mathbf{B}'$ has the same expression as the manifold of a continuous virtual ULA. The rebuilt vector $\mathbf{z}'$ can be expressed as

$$\mathbf{z}' = \mathbf{B}' \cdot \mathbf{p} + \mathbf{n}'. \tag{20}$$

Since the new source signal $\mathbf{p}$ and the new noise vector $\mathbf{n}'$ are no longer incoherent, we use spatial smoothing technique

[5] to build the rank of a positive semi-definite matrix from this new model. We divide the new received data vector $\mathbf{z}'$ into multiple vectors $\mathbf{z}_i'$ so that its corresponding virtual ULA array is divided into multiple overlapping sub-arrays. Then we compute the autocorrelation-like matrix of each divided received data vector $\mathbf{z}_i'$

$$\mathbf{R}_{zi} \triangleq \mathbf{z}_i' \mathbf{z}_i'^H \tag{21}$$

Taking the average of the autocorrelation matrices of all sub-arrays, we can get the final spatial smoothed matrix $\mathbf{R}_{zz}$ as

$$\mathbf{R}_{zz} = \frac{1}{DOF} \sum_{i=1}^{DOF} \mathbf{R}_{zi} \tag{22}$$

where DOF equals the number of sub-arrays and denotes the maximum number of detectable sources.

Finally, we can accomplish DOA estimation by applying MUSIC algorithm [4] on $\mathbf{R}_{zz}$.

5. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our proposed sum and difference co-prime array (SD-CPA) technique through simulations over matlab. We run MUSIC algorithm to detect the DOAs of a group of uniformly distributed sources. We compare the performance of SD-CPA with those of three other reference methods which exploit the dual-frequency co-prime array (DF-CPA), the extended co-prime array (E-CPA) and uniform linear array (ULA) respectively.

In the first study, we consider a conventional co-prime array configuration with $M = 4$ and $N = 5$, so the total number of physical sensors is $M + N - 1 = 8$. The total number of sum and difference items generated is $2MN + 2M + 2N - 1 = 57$. After using the spatial smoothing technique, the available DOF is $MN + M + N = 29$. We generate 25 sinusoidal sources with $SNR = 0dB$, and angles θ_i uniformly distributed within the range $-60°$ to $60°$. The covariance matrix is estimated using 2000 snapshots. From the MUSIC spectrum in Fig. 6, we can see that SD-CPA can identify all sources in the spectrum, which demonstrates its effectiveness in identifying a larger number of sources using a much smaller number of physical sensors.

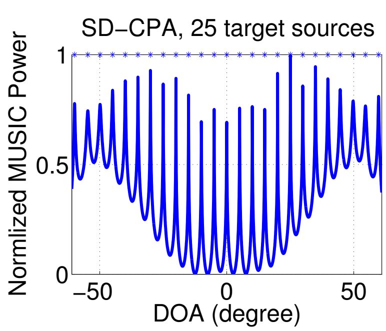
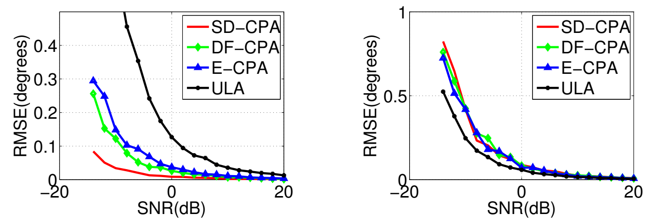


Fig. 6: MUSIC spectrum

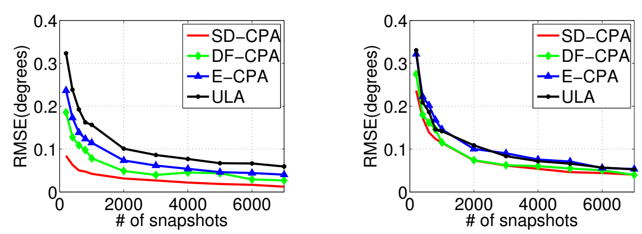
We then compare the root mean squared error (RMSE) of the DOA estimation among the four different methods, with the covariance matrices estimated using 2400 snapshots and the number of target sources set to $D = 9$. Fig. 7 (a) shows the result when all methods use the same number of physical sensors, which is 12. Compared to DF-CPA, E-CPA and ULA, our proposed method reduces the RMSE over 50%, 60% and 90%, respectively. When using the same number of physical sensors, our method has much larger number of DOFs. Thus, when detecting the same number of sources, our method has a better detection performance. Fig. 7 (b) shows the comparison result when all methods have the same or similar number of DOFs, and the numbers of physical sensors are respectively 6, 9, 9 and 13. We can see that ULA has the best performance, because all other methods use the ideal virtual ULA which still suffers from some information loss. Compared to the peer schemes, our method has a comparable detection performance but uses the fewest physical sensors.



(a) Same number of sensors (b) Same DOF level

Fig. 7: RMSE versus SNR

In Fig. 8, we investigate the impact of snapshots instead of SNR. All conditions are same as the previous study, except that we vary the number of snapshots and the SNR is set to $5dB$. Fig. 8 (a) shows that, when using the same number of physical sensors and under the same number of snapshots, our method reduces RMSE over 50%, 60% and 70% compared to those of DF-CPA, E-CPA and ULA. In other words, if we want to achieve the same RMSE, our method can save over 60%, 75% and 90% snapshots. Our method thus can have much faster estimation speed while reducing the cost. The reason is the same as that in the previous study, that our method has a larger DOF. Fig. 8 (b) shows our method has the detection performance comparable to DF-CPA and E-CPA, however we use much fewer number of physical sensors.



(c) Same number of sensors (d) Same DOF level

Fig. 8: RMSE versus number of snapshots

Finally, we compare the maximum number of detectable sources using the four different methods, each having 12 physical sensors. The threshold of RMSE is set to $T = 0.5$ and the number of snapshots is 2000. From Fig. 9, we can see that our method can detect over 1.5 times, 2 times and 3 times the number of sources compared to DF-CPA, E-CPA and ULA method, respectively.

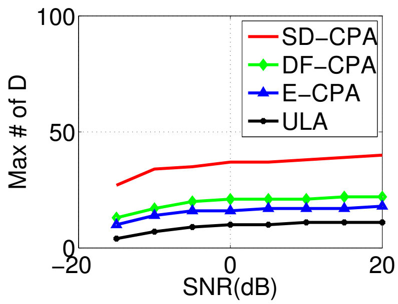


Fig. 9: Maximum number of detectable source

6. CONCLUSION

We have proposed a technique SD-CPA which concurrently exploits the sum and different co-array to form a larger size of continuous virtual array using only the conventional co-prime array infrastructure. The proposed method enhances the maximum number of detectable sources and improves the detection performance. More importantly, it reduces the number of required physical sensors thus the array aperture, and the number of required snapshots thus increasing the detection speed. This shows the high efficiency of our method in both space domain and temporal domain.

7. REFERENCES

[1] A Moffet, “Minimum-redundancy linear arrays,” *Antennas and Propagation, IEEE Transactions on*, vol. 16, no. 2, pp. 172–175, Mar 1968.

[2] P. Pal and P.P. Vaidyanathan, “Nested arrays: A novel approach to array processing with enhanced degrees of freedom,” *Signal Processing, IEEE Transactions on*, vol. 58, no. 8, pp. 4167–4181, Aug 2010.

[3] P.P. Vaidyanathan and P. Pal, “Sparse sensing with co-prime samplers and arrays,” *Signal Processing, IEEE Transactions on*, vol. 59, no. 2, pp. 573–586, Feb 2011.

[4] R.O. Schmidt, “Multiple emitter location and signal parameter estimation,” *Antennas and Propagation, IEEE Transactions on*, vol. 34, no. 3, pp. 276–280, Mar 1986.

[5] Tie-Jun Shan, Mati Wax, and Thomas Kailath, “On spatial smoothing for direction-of-arrival estimation of coherent signals,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 4, pp. 806–811, 1985.

[6] P. Pal and P.P. Vaidyanathan, “Coprime sampling and the music algorithm,” in *Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE*, Jan 2011, pp. 289–294.

[7] S. Qin, Y. D. Zhang, and M. G. Amin, “Doa estimation exploiting coprime frequencies,” in *Proc. SPIE 9103, Wireless Sensing, Localization, and Processing IX*, May 2014.

[8] Si Qin, Y.D. Zhang, and M.G. Amin, “Generalized coprime array configurations for direction-of-arrival estimation,” *Signal Processing, IEEE Transactions on*, vol. 63, no. 6, pp. 1377–1390, March 2015.

[9] J. Ramirez and J. Krolik, “Multiple source localization with moving co-prime arrays,” in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, April 2015, pp. 2374–2378.

[10] E. BouDaher, F. Ahmad, and M.G. Amin, “Sparsity-based direction finding of coherent and uncorrelated targets using active nonuniform arrays,” *Signal Processing Letters, IEEE*, vol. 22, no. 10, pp. 1628–1632, Oct 2015.