A search-free DOA estimation algorithm for coprime arrays

CrossMark

Zhiyuan Weng, Petar M. Djurić *

Stony Brook University, Stony Brook, New York 11794, United States

A R T I C L E   I N F O

A B S T R A C T

Recently, coprime arrays have been in the focus of research because of their potential in exploiting redundancy in spanning large apertures with fewer elements than suggested by theory. A coprime array consists of two uniform linear subarrays with inter-element spacings $M\lambda/2$ and $N\lambda/2$, where M and N are coprime integers and λ is the wavelength of the signal. In this paper, we propose a fast search-free method for direction-of-arrival (DOA) estimation with coprime arrays. It is based on the use of methods that operate on the uniform linear subarrays of the coprime array and that enjoy many processing advantages. We first estimate the DOAs for each uniform linear subarray separately and then combine the estimates from the subarrays. For combining the estimates, we propose a method that projects the estimated point in the two-dimensional plane onto one-dimensional line segments that correspond to the entire angular domain. By doing so, we avoid the search step and consequently, we greatly reduce the computational complexity of the method. We demonstrate the performance of the method with computer simulations and compare it with that of the FD-root MUSIC method.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Direction-of-arrival (DOA) estimation using sensor arrays is a problem that is frequently encountered in many engineering areas including radar, sonar, and wireless communication, and it has been studied for several decades. Recently, the notion of coprime array signal processing has emerged as an area of interest where, as the name suggests, the processing is applied to signals acquired by coprime arrays [1–4]. In general, it has been shown that coprime sampling allows us to sample a signal sparsely and estimate parameters of signals at higher resolutions [2]. The parameters can be the frequencies of temporal signals or the DOAs of spatial signals. By sparse sampling, one can reduce the rate of analog-to-digital converter (ADC) for sampling in the temporal domain [4] or reduce the number of array sensor elements for sampling in the spatial domain [1]. This entails that systems where coprime sampling is applied can have lower cost than systems with Nyquist sampling, yet without degrading the performance of the latter. For example, the cost of ADC usually grows exponentially with the bandwidth or the sampling rate. By using coprime sampling, one can reduce the requirements on the analog front-end by significantly lowering the sampling rate [4].

In general, the idea behind coprime arrays is to extend the concept of minimally redundant sensor arrays and span large apertures by using far fewer elements than prescribed by textbook antenna theory. A coprime array can be constructed with two uni-

form linear arrays with M and N elements with inter-element spacings $M\lambda/2$ and $N\lambda/2$, respectively, and where M and N are coprime integers, i.e., integers whose greatest common divisor is one. It has been shown that in the presence of spatially stationary fields, one can match the information extracted with such arrays to that of fully populated arrays with MN elements [2]. As already pointed out, a very important payoff for the use of coprime arrays is simplified and reduced cost array designs. Instead of having an array with MN elements, one can use an array with $M + N - 1$ elements (where one element is shared).

The reduced cost of coprime arrays represents a strong motivation for studying DOA methods that can process signals for such arrays. An obvious approach is to consider methods that are developed to operate on arbitrary geometry arrays. In this class there are several popular DOA estimation methods including beamspan [5], Capon [6], MUSIC [7] and several versions of maximum likelihood estimators [8]. For finding the optimal estimate, however, they all require a computationally expensive search step in a nonconvex space. We note that for uniform linear arrays (ULAs), the array response vector has a Vandermonde form, which allows the search step to be replaced by polynomial rooting. These so-called *search-free* algorithms include IQML [9], MODE (Method of Direction Estimation) [10], root-MUSIC [11] and ESPRIT [12]. An interesting alternative to the methods for arbitrary arrays are the approaches that transform the arbitrary arrays to equivalent ULAs so that one can take the advantage of efficient algorithms for ULAs, e.g., array interpolation [13], manifold separation [14] and Fourier-Domain (FD) root-MUSIC [15]. These approaches basically try to approximate the steering vectors by virtual ULAs. Their drawback

is that for achieving satisfactory performance, they require a large number of virtual arrays, which entails increased complexity. Furthermore, they suffer fixed error at high signal-to-noise ratios (SNRs) and thus, do not converge to the Cramér–Rao bound as the SNR increases.

The DOA estimation with ULAs has been thoroughly studied, but there has not been much work for designing special algorithms that take advantage of the *partial* uniform linear structure of coprime arrays. In [1], the authors apply the MUSIC algorithm to coprime processing, where they mainly consider the problem from the perspective of degree of freedom, i.e., how many signals the coprime array can identify. To achieve additional degree of freedom with coprime arrays, that is, to enhance the rank of the covariance matrix of the received signal, they propose the use of spatial smoothing. Nevertheless, the proposed method is not search-free, and its accuracy and computational complexity have not been investigated in detail. In [4], where processing of time series is addressed, the authors use an iterative method to search for the optimal frequency. Although it is search-free, it does not use the coprime property and it cannot handle the case of multiple signals.

In this paper, we address the problem of DOA estimation with coprime arrays with the emphasis on reduced computational complexity while preserving estimation accuracy. We propose a search-free method that exploits the uniform linear structure of the subarrays of the coprime arrays. Considering that the DOA estimation using ULAs is fast and accurate, we first estimate the DOA for each subarray separately. Since the inter-element distance for each subarray is larger than half-wavelength of the signal, ambiguity is present and as a result, the DOAs cannot be uniquely determined. To resolve the ambiguity, we use a projection-like method that combines the estimates from different subarrays, where the correctness of the estimate is guaranteed as a consequence of the coprimeness.

The paper is organized as follows. In Section 2, we introduce the model and briefly review the DOA estimation with ULAs. The proposed algorithm is discussed in detail in Section 3 and the numerical experiments are presented in Section 4. Section 5 provides final remarks.[1]

We use the following notation: $(\cdot)^{\top}$ and $(\cdot)^{H}$ denote the transpose and the conjugate transpose, respectively; $\mathbb{C}$ refers to the complex space; $\delta_{k,l}$ signifies the Kronecker delta; $\mathbb{E}(\cdot)$ stands for the expectation operator; and tr$\{\mathbf{A}\}$ is the trace of the matrix $\mathbf{A}$.

2. DOA estimation for ULA

The problem of finding the DOAs of D narrowband plane waves impinging on a ULA of L sensors can be modeled as follows [16]:

$$\mathbf{y}(k) = \mathbf{A}\mathbf{x}(k) + \mathbf{w}(k), \quad k = 1, \ldots, K, \tag{1}$$

where $\mathbf{y}(k) \in \mathbb{C}^{L \times 1}$ is the signal received by the sensors at the kth time slot; K is the number of snapshots; $\mathbf{x}(k) \in \mathbb{C}^{D \times 1}$ contains the complex envelopes of the emitter signals; $\mathbf{w}(k)$ is a noise process; and

$$\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_D] \in \mathbb{C}^{L \times D} \tag{2}$$

with

$$\mathbf{a}_d = \left[1, \exp(jp_1\psi_d), \ldots, \exp(jp_{L-1}\psi_d)\right]^{\top} \in \mathbb{C}^{L \times 1}$$

$$d = 1, \ldots, D; \tag{3}$$

$p_l \lambda/2$ is the distance from the lth sensor to the reference point; ψ_d is the DOA of our interest. The number of sources, D, is assumed

to be known. Thus, $\Psi = [\psi_1, \ldots, \psi_D]$ is the vector of DOAs we wish to estimate. We assume the signal $\mathbf{x}(k)$ and the noise $\mathbf{w}(k)$ are independent zero-mean complex Gaussian random processes with the following moments:

$$\mathbb{E}\left[\mathbf{x}(k)\mathbf{x}^{H}(l)\right] = \mathbf{P}\delta_{k,l}, \qquad \mathbb{E}\left[\mathbf{x}(k)\mathbf{x}^{T}(l)\right] = \mathbf{0}, \quad k \neq l, \tag{4}$$

$$\mathbb{E}\left[\mathbf{w}(k)\mathbf{w}^{H}(l)\right] = \sigma^2 \mathbf{I}\delta_{k,l}, \qquad \mathbb{E}\left[\mathbf{w}(k)\mathbf{w}^{T}(l)\right] = \mathbf{0}, \quad k \neq l \tag{5}$$

with $\mathbf{I}$ being the identity matrix, $\mathbf{P}$, the signal covariance matrix, and σ^2, the noise power. The covariance matrix of the received signal $\mathbf{y}(k)$ can be written as

$$\mathbf{R} = \mathbf{A}\mathbf{P}\mathbf{A}^{H} + \sigma^2 \mathbf{I}. \tag{6}$$

We denote by $\hat{\mathbf{R}}$ the sample covariance matrix

$$\hat{\mathbf{R}} = \frac{1}{K}\sum_{k=0}^{K-1} \mathbf{y}(k)\mathbf{y}^{H}(k). \tag{7}$$

The eigendecomposition of $\hat{\mathbf{R}}$ in (7) can be written as

$$\hat{\mathbf{R}} = \hat{\mathbf{E}}_s \hat{\mathbf{\Lambda}}_s \hat{\mathbf{E}}_s^{H} + \hat{\mathbf{E}}_n \hat{\mathbf{\Lambda}}_n \hat{\mathbf{E}}_n^{H}, \tag{8}$$

where $\hat{\mathbf{\Lambda}}_s \in \mathbb{C}^{D \times D}$ and $\hat{\mathbf{\Lambda}}_n \in \mathbb{C}^{(L-D) \times (L-D)}$ are diagonal matrices that contain the eigenvalues of the signal and the noise subspaces, respectively, whereas $\hat{\mathbf{E}}_s \in \mathbb{C}^{L \times D}$ and $\hat{\mathbf{E}}_n \in \mathbb{C}^{L \times (L-D)}$ are composed of the eigenvectors of the signal and the noise subspaces, respectively.

For ULAs, the steering vector becomes

$$\mathbf{a}_i(\psi_i) = \left[1, \exp(j\psi_i), \ldots, \exp(j(L-1)\psi_i)\right]^{\top}. \tag{9}$$

There exist a large number of fast algorithms for estimating DOAs with ULAs, among which MODE is the leading candidate [17,18]. Since we later use MODE as part of our algorithm, we review it here briefly. The MODE algorithm was originally proposed in [10]. It estimates the DOAs via the minimization of the following function:

$$\mathbf{b} = \underset{\mathbf{b} \in \mathbb{C}^{D \times 1}}{\arg \min} \operatorname{tr}\left\{\mathbf{\Pi}_{\mathbf{B}} \hat{\mathbf{E}}_s \mathbf{W} \hat{\mathbf{E}}_s^{H}\right\} \tag{10}$$

where

$$\mathbf{b} = [b_0, b_1, \ldots, b_D]^{\top}, \tag{11}$$

$$\mathbf{\Pi}_{\mathbf{B}} = \mathbf{B}(\mathbf{B}^{H}\mathbf{B})^{-1}\mathbf{B}^{H}, \tag{12}$$

$$\mathbf{W} = (\hat{\mathbf{\Lambda}}_s - \hat{\sigma}^2 \mathbf{I})^2 \hat{\mathbf{\Lambda}}_s^{-1}, \tag{13}$$

$$\hat{\sigma}^2 = \frac{1}{L-D}\operatorname{tr}\{\hat{\mathbf{\Lambda}}_N\}, \tag{14}$$

and $\mathbf{B}$ is an $L \times (L-D)$ Toeplitz matrix defined by

$$\mathbf{B}^{H} = \begin{bmatrix} b_D & \cdots & b_0 & 0 & \cdots & 0 \\ 0 & b_D & \cdots & b_0 & \cdots & 0 \\ \vdots & \ddots & \ddots & & \ddots & \vdots \\ 0 & \cdots & 0 & b_D & \cdots & b_0 \end{bmatrix}. \tag{15}$$

Let $z = \exp(j\psi_i)$, and denote by $b(z)$ the complex-valued polynomial

$$b(z) = b_0 z^D + b_1 z^{D-1} + \cdots + b_D \tag{16}$$

$$= b_0 \prod_{i=1}^{D}(z - z_i). \tag{17}$$

The angles of the roots of the polynomial are the estimates of the DOAs. It was shown in [10] that the MODE algorithm is an asymptotically efficient estimator of the DOAs. As per [17,18], the convergence of MODE is guaranteed. To minimize (10), both iterative and non-iterative steps have been proposed. See [17,18] for details.

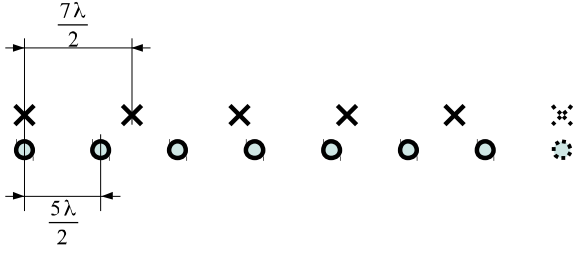[1] An earlier version of this paper can be found at http://arxiv.org/abs/1301.4155.

Fig. 1. A coprime array with two ULAs, where $M = 7$ and $N = 5$.

3. DOA estimation for coprime arrays

We consider a coprime array with two uniform linear subarrays. We assume that the two subarrays have inter-element spacing $M\lambda/2$ and $N\lambda/2$, respectively, with M and N being coprime. Fig. 1 shows the case for $N = 7$ and $M = 5$. Because the two subarrays share the first sensor, the total number of sensors L is equal to $M + N - 1$. The corresponding steering vector in (3) becomes

$$\mathbf{a}_d = \big[1, e^{j\psi_d M}, e^{j\psi_d 2M}, \ldots, e^{j\psi_d(N-1)M}$$
$$e^{j\psi_d N}, e^{j\psi_d 2N}, \ldots, e^{j\psi_d(M-1)N}\big]^\top. \qquad (18)$$

For such arrays, the efficient methods for ULAs are not directly applicable. We can of course use the algorithms for arbitrary arrays, but they are slow and the uniform linear structure of the subarrays is wasted. Our objective is to develop an algorithm that is applicable to coprime arrays while at the same time enjoys the efficiency brought by the uniform linear structure of the subarrays. The proposed approach is composed of two steps. We first estimate the DOAs from each subarray separately, which is similar to the approach in [4]. We then combine the two estimates in an innovative way.

3.1. DOA estimation using a single subarray

For a subarray, the inter-element spacing is $N\lambda/2$ with $N > 1$. This has the same effect as we undersample a signal by a factor of N in the temporal domain. According to the basic sampling theorem, we will have ambiguities, or aliasing, in the angular domain. Specifically, if we use the same algorithm as described above, the polynomial in (17) becomes

$$b'(z) = b_0' \prod_{i=1}^{D} (z^N - z_i'). \qquad (19)$$

This polynomial has $N \times D$ roots instead of D. If z_i is a root of (17), it is still a root of (19). The problem is that aliasing occurs with the period of $2\pi/N$; $z_i \exp(jn\frac{2\pi}{N})$ for $n = 1, \ldots, N - 1$ are also the roots of (19), as shown in Fig. 2. Therefore, we have no way of distinguishing the z_i we want from the remaining $N - 1$ roots.

Note that in the first step, it does not matter which algorithm we use. In the experiment, we selected MODE, which was shown to have the best performance for ULAs [18].

3.2. Combination of the estimates from subarrays

The present ambiguities in the estimates from the two subarrays notwithstanding, we show that we are able to disambiguate them. For example, in the case of one source and no noise, in Fig. 2, we can see that the estimates from the two subarrays coincide at the red arrows. Therefore, we can be very confident that the position of the red arrow is the DOA. In the presence of noise, however, coincidences as the one in Fig. 3 are unlikely, and consequently, it becomes difficult to tell the exact DOA. Furthermore,
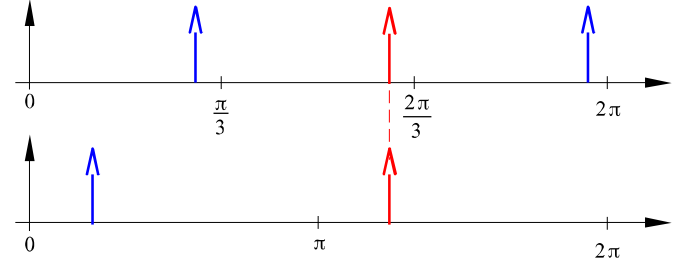

Fig. 2. Ambiguity on the angular domain, case 1. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)
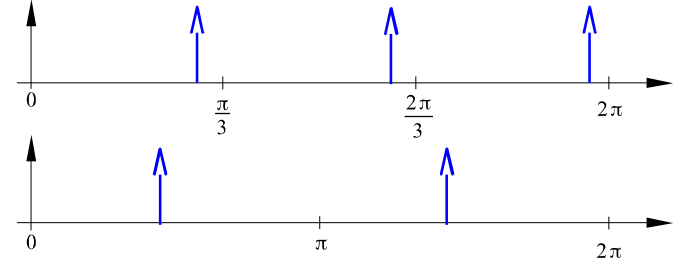

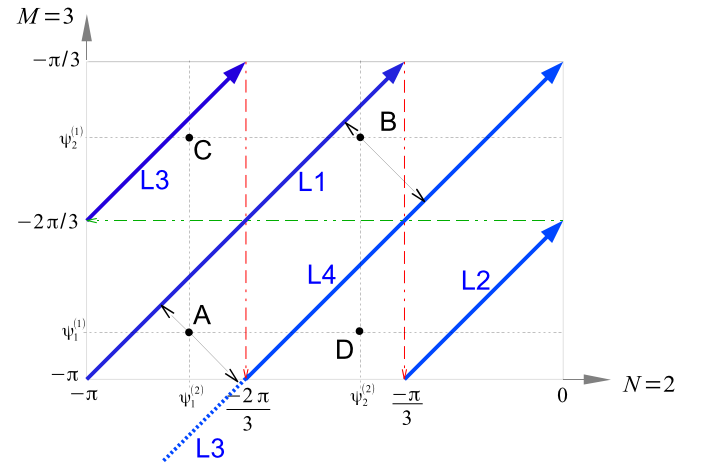**Fig. 3.** Ambiguity on the angular domain, case 2.

Fig. 4. Projecting the point specified by the two estimates onto the entire angular domain. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

in the case with multiple sources, obtaining the DOAs in this way becomes impossible.

We propose a projection method that can uniquely determine the exact DOA. Recall that the aliasing period in the angular domain for the subarray is $2\pi/N$. Thus, we only need to take the values between $[-\pi, -\pi + 2\pi/N]$. The ranges of the outputs of the two subarrays define a rectangle in the two-dimensional plane as shown in Fig. 4, where $N = 2$ and $M = 3$. It is not difficult to see that the 45 degree oblique line segments colored in blue correspond to the entire angular domain. The Chinese remainder theorem guarantees that the map is one-to-one and onto as a result of the coprimeness. This is the key point of our algorithm. In Fig. 4, the entire angular domain consists of four line segments; L1 corresponds to $[-\pi, -\pi/3]$; L2, $[-\pi/3, 0]$; L3, $[0, \pi/3]$; and L4, $[\pi/3, \pi]$. For general M and N, there will be $M + N - 1$ oblique line segments. Suppose the outputs of the first and the second subarray are $\psi^{(1)}$ and $\psi^{(2)}$, respectively. The two estimates specify a point in the plane. We project the point onto the oblique line segments that corresponds to the entire angular domain. Specifically,

we seek the point on the oblique line segments that is nearest to that point specified by the two estimates. This ensures that the combination is optimal and the result is valid. For point **B** in Fig. 4, we can simply draw a line that is vertical to the oblique line and measure the distance from **B** to the intersections. For point **A**, however, one intersection falls outside the rectangle area and should be wrapped around. As a result, the intersection is actually on L3 in the figure. After the nearest point on the line segments is found, it is easy to calculate the corresponding DOA by solving a set of modular linear equations. As an alternative to solving the equations, one may have a precalculated lookup table with solutions of the modular equations.

The main difference between the proposed method and the one in [4] is that here we take advantage of the coprimeness and use a projection process in the search for optimal DOAs, while the method in [4] iteratively tries different combinations of the estimates and find the best one among them. Although solving the modular equations in our method requires an iterative process, it can be avoided by using a lookup table.

3.3. Combination of the estimates in the case of multiple sources

In the case of multiple sources, the method gets more complicated. Suppose that the outputs of the first subarray are $\psi_1^{(1)}$ and $\psi_2^{(1)}$ and the outputs of the second subarray, $\psi_1^{(2)}$ and $\psi_2^{(2)}$. We have no way of knowing which value corresponds to the first and which to the second target. Therefore, we have to check all the four points. In Fig. 4, $\psi_1^{(1)}$, $\psi_2^{(1)}$, $\psi_1^{(2)}$ and $\psi_2^{(2)}$ define four points on the two-dimensional plane, which are marked **A**, **B**, **C** and **D**. Once the candidate points are obtained, we can readily evaluate the likelihood of these points and take the best D ones.

In short, the proposed algorithm can be summarized as follows:

1. Estimate the D angles for each subarray.
2. Plot the corresponding D^2 points on the 2-dimensional plane.
3. Project these points to the nearest diagonal line segments.
4. Map the projected D^2 points back onto the entire angular domain.
5. Evaluate the likelihood at the D^2 points.
6. Select D points with the largest likelihood.

3.4. Example

In this subsection, we provide a simple example to better illustrate the proposed method. We consider the case for $N = 2$, $M = 3$ and the number of targets $D = 2$. Suppose the outputs of the first subarray are $\psi_1^{(1)} = -2.6\frac{\pi}{3}$ and $\psi_2^{(1)} = -1.8\frac{\pi}{3}$, and those of the second subarray are $\psi_1^{(2)} = -2.5\frac{\pi}{3}$ and $\psi_2^{(2)} = -1.25\frac{\pi}{3}$. We denote by $\psi_1^{(0)}$ and $\psi_2^{(0)}$ the final results of the estimation. Using the estimates of the subarrays, we can locate the following four points on the two-dimensional plane: **A**$(-2.5\frac{\pi}{3}, -2.6\frac{\pi}{3})$, **B**$(-1.25\frac{\pi}{3}, -1.8\frac{\pi}{3})$, **C**$(-2.5\frac{\pi}{3}, -1.8\frac{\pi}{3})$ and **D**$(-1.25\frac{\pi}{3}, -2.6\frac{\pi}{3})$. Also, we can see by simple calculation that

- **A** is closer to L1 with distance $\frac{0.1\pi}{3\sqrt{2}}$;
- **B** is closer to L4 with distance $\frac{0.45\pi}{3\sqrt{2}}$;
- **C** is closer to L3 with distance $\frac{0.3\pi}{3\sqrt{2}}$;
- **D** is closer to L4 with distance $\frac{0.35\pi}{3\sqrt{2}}$.

We then project these points back onto the diagonal line segments. For example, **B** is projected onto L4 and the projected point, say **B**$_L$, is $(-1.025\frac{\pi}{3}, -2.025\frac{\pi}{3})$. Subsequently, we construct the modular equations
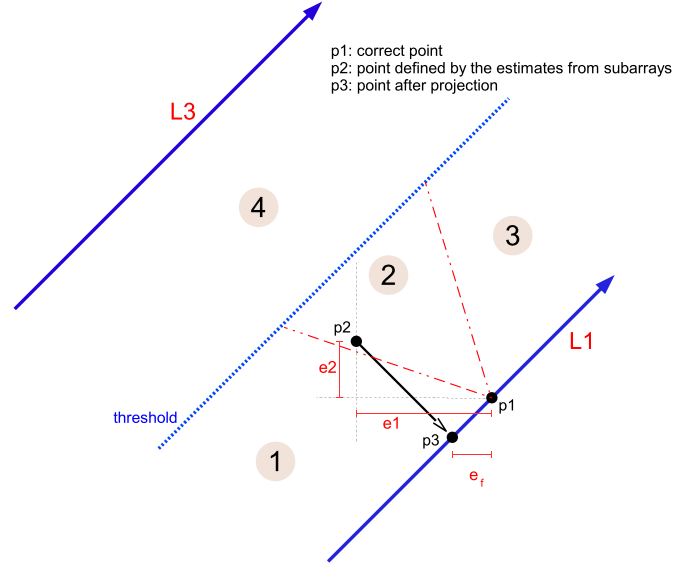

Fig. 5. Illustration of projection error.

$$\psi^{(0)} \bmod 2 = -1.025\frac{\pi}{3}, \tag{20}$$

$$\psi^{(0)} \bmod 3 = -2.025\frac{\pi}{3}, \tag{21}$$

and the solution will be the value for **B**$_L$ on the entire angular domain. The equations can be solved using the extended Euclidean algorithm. Alternatively, as we stated before, we can also use a lookup table to solve the equations. After we obtain the four values for the four points, we can then evaluate the likelihood at each point and take the largest two values as the final estimates.

3.5. Discussion

3.5.1. Asymptotic performance

In this subsection, we briefly discuss the asymptotical performance of the proposed algorithm. Recall that our method basically consists of two steps. First, we estimate the DOA separately for each subarray, and second, we combine the estimates by projecting the combined point onto a predefined diagonal line in the two-dimensional plane defined by the two subarrays. For the first step, any classical DOA estimation method can be used. In our simulations, we applied the MODE method, which is a type of maximum likelihood estimator [18].

It is well known that maximum likelihood estimates are asymptotically unbiased and normally distributed. The asymptotical property of the proposed algorithm depends on the SNR level. For high SNRs, the point on the plane defined by the two estimates of the subarrays will probably lie very close to the correct point. After projection, it could be even closer. This can be seen in Fig. 5. The estimation errors from the subarrays are indicated by e_1 and e_2. In the second step, we project the point back onto the line segment and obtain $p3$. We can see that the distance between $p1$ and $p3$ is shorter than the one between $p1$ and $p2$. It is obvious that as long as $p2$ does not lie outside the blue threshold, which is the line between the two line segments, $p3$ is always closer to $p1$ than $p2$ is to $p1$. Moreover, we notice that the projection is linear. Since the estimates from the subarrays are asymptotically unbiased, we can expect that the final estimate is also unbiased. In other words, the error would not propagate for high SNRs.

Under low SNRs, however, $p2$ is more likely to fall on the other side of the blue threshold. As a result, $p2$ will be incorrectly projected to a different line segment and therefore, the final estimate

will be poor. This explains why a threshold effect can be seen in the simulations. We point out that this effect is seen in many other unwrapping phase-based methods. After the SNR goes above certain level, the mean square error (MSE) decreases rapidly. We defined four areas in Fig. 5. Given $p1$ as shown in the figure, if the point $p2$ lies in Area 1, we have $e_2 < e_f < e_1$; for Area 2, we have $e_f < e_1$ and $e_f < e_2$; for Area 3, we have $e_1 < e_f < e_2$; for Area 4, $p2$ is projected to the other line segment, and therefore the error is magnified. Then we have $e_1 < e_f$ and $e_2 < e_f$.

3.5.2. Maximum number of detectable sources

For ULAs, it is well known that an array with N elements can identify up to $N-1$ targets. Thus, the two subarrays in the coprime array can identify $N-1$ and $M-1$ targets, respectively. Consequently, there will be $(N-1)(M-1)$ intersections on the two-dimensional plane, which entails that the maximum number of sources that a coprime array can identify is $(N-1)(M-1)$.

We point out that our algorithm is particularly suitable for the case of a single target. For multiple sources, it may produce poor results because of the resolution problem in the processing step of each subarray. For ULAs, when two sources are very close to each other, the estimates may be poor. For coprime arrays, it is very likely that the true values of the sources overlap on the wrapped angular domain of one of the subarrays even if on the entire angular domain they do not. In that case, the estimates from the subarray are not accurate at all, and consequently the final estimates are unreliable.

4. Simulations

In this section, we use numerical experiments that demonstrate the performance of the proposed method. We show results of simulations that compare the method with that of the FD root-MUSIC method as well as with the Cramér–Rao bound (CRB). We note that the FD root-MUSIC method uses Fourier series coefficients to approximate the null-spectrum function [15], and we use it because it was shown to have better performance than other search-free methods [19].

In the simulations, we used two sets of arrays, one with $N = 5$, $M = 7$, and one with $N = 29$, $M = 31$ elements. For each coprime arrays, we test the cases of one source and three sources.

4.1. The case of one source

First, we simulated one source with $\psi_1 = 0.1\pi$. In the first two experiments, we studied the performance of the method on the coprime array $N = 5$, $M = 7$. In the first of them, we fixed the number of snapshots to $K = 100$, and we varied the SNR from -30 dB to 10 dB. For the FD root-MUSIC, we used $S = 20, 50, 100$, and 200 Fourier series coefficients, respectively. We computed the mean square error (MSE) of the methods by using 2000 independent realizations. The results and the CRB are presented in Fig. 6. We can see that at SNRs below -8 dB, the FD root-MUSIC outperforms the proposed method. For SNRs greater than -8 dB, the performance of the proposed method approaches the CRB while the FD root-MUSIC suffers from fixed errors due to its approximations. As the number of Fourier series coefficients increases, as expected, the performance of the FD root-MUSIC method improves. However, the method cannot reach the CRB.

In the second experiment, we kept the SNR at -13 dB, and varied the number of snapshots from 100 to 1000 in steps of 100. The remaining parameters of the experiment were the same as before. From Fig. 7 we see that when the number of snapshots is 100, the MSE of the proposed method is far from the CRB. The results are shown in Fig. 7. We see that the FD root-MUSIC is better than our method when the number of snapshots $K = 600$ or smaller. On
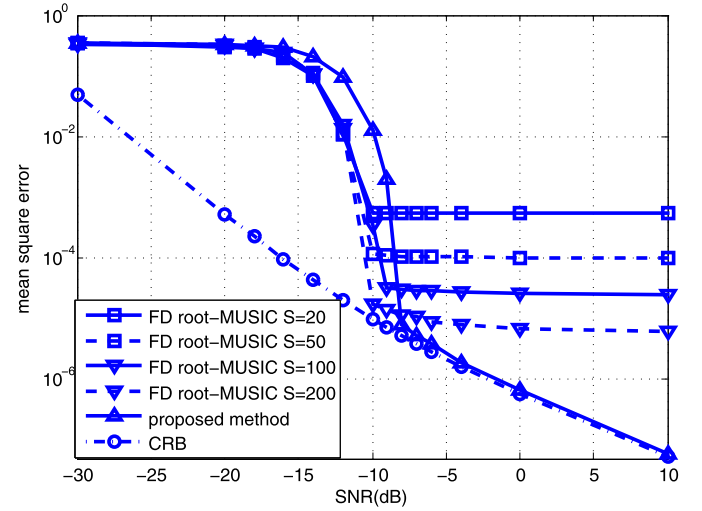

Fig. 6. Mean square error performance at different SNRs for one source ($K = 100$). The coprime array is of size $(5, 7)$.
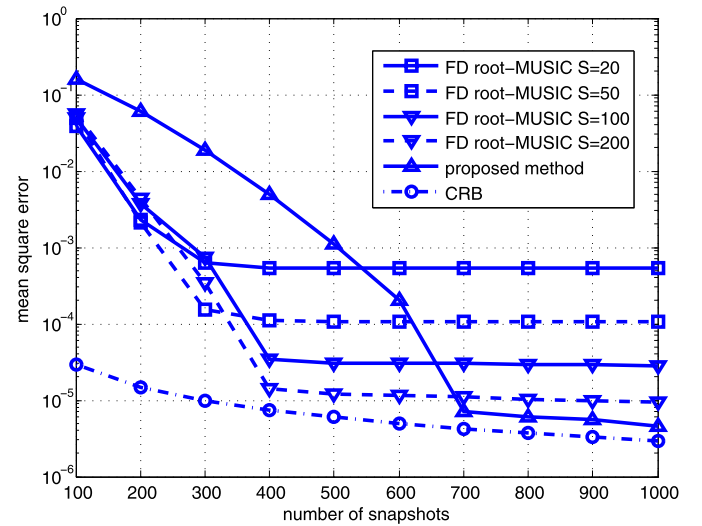

Fig. 7. Mean square error performance for different number of snapshots for one source (SNR $= -13$ dB). The coprime array is of size $(5, 7)$.

the other hand, our method approaches the CRB when $K > 600$ whereas the FD root-MUSIC does not improve for $K > 400$.

In the remaining two experiments, we repeated the simulations with a coprime array with $N = 29$, $M = 31$ elements. Again, first we kept $K = 100$ and varied the SNR from -30 dB to 10 dB. The results of the MSE's and the CRB are shown in Fig. 8. We observe similar patterns as in Fig. 6. However, now the proposed method reaches the CRB at -12 dB. For SNRs greater than -12 dB, the difference in performance between the two methods is by orders of magnitude and in favor of the proposed method.

Finally, in the last experiment, we kept the SNR at -13 dB, and varied the number of snapshots from 100 to 1000, in steps of 100. The MSEs of the methods and the CRB are displayed in Fig. 9. We observe, that our method follows the CRB as the number of snapshots increases, whereas the MSE of FD root-MUSIC remains constant.

4.2. The case of three sources

In this subsection, we repeated the simulations under the presence of three targets. We set the angles to be $\psi_1 = -0.4\pi$, $\psi_2 = 0.25\pi$, $\psi_3 = 0.33\pi$. Similarly we show four figures here.
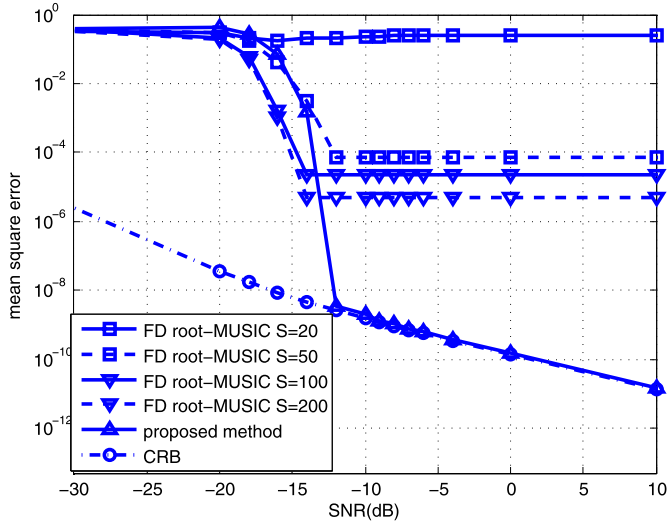
Fig. 8. Mean square error performance at different SNRs for one source ($K = 100$). The coprime array is of size $(29, 31)$.
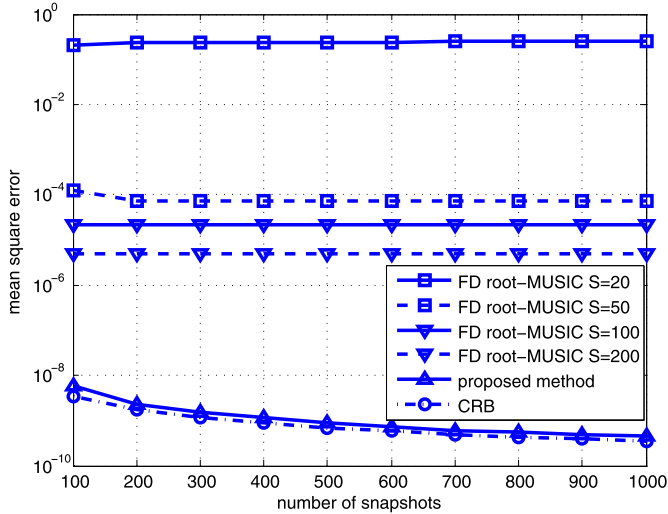

Fig. 9. Mean square error performance for different number of snapshots for one source (SNR $= -13$ dB). The coprime array is of size $(29, 31)$.

Fig. 10. Mean square error performance at different SNRs for three sources ($K = 100$). The coprime array is of size $(5, 7)$.

Fig. 11. Mean square error performance for different number of snapshots for three sources (SNR $= -8$ dB). The coprime array is of size $(5, 7)$.

The results of the coprime array with $N = 5$, $M = 7$ are displayed in Figs. 10 and 11. Fig. 10 presents the MSEs for different values of SNR with $K = 100$. We observe similar patterns as in Figs. 6 and 8. The proposed method reaches the CRB at about -5 dB. For SNRs greater than -7 dB, the difference in performance between the two methods is by orders of magnitude and again in favor of the proposed method. Also we notice that the MSEs of the FD root-MUSIC methods with $S = 50, 100, 200$ are almost the same. That is, with the increase of the approximation order, the performance of the method does not improve. Fig. 11 shows the MSEs for different values of K with SNR $= -8$ dB. Our method follows the CRB when the number of snapshots is larger than 400, whereas the MSEs of FD root-MUSIC remains constant. The results of the coprime array with $N = 29$, $M = 30$ are shown in Figs. 12 and 13. Similarly, Fig. 12 shows the MSEs for different values of SNR with $K = 100$. The proposed method outperforms the FD root-MUSIC methods under both low and high SNRs. This is because for arrays with large size, the number of 200 Fourier coefficients is far from enough to provide good approximation. Fig. 13 shows the MSEs for different values of K with SNR $= -8$ dB. Our method follows the CRB when the number of snapshots is larger than 900, and its per-
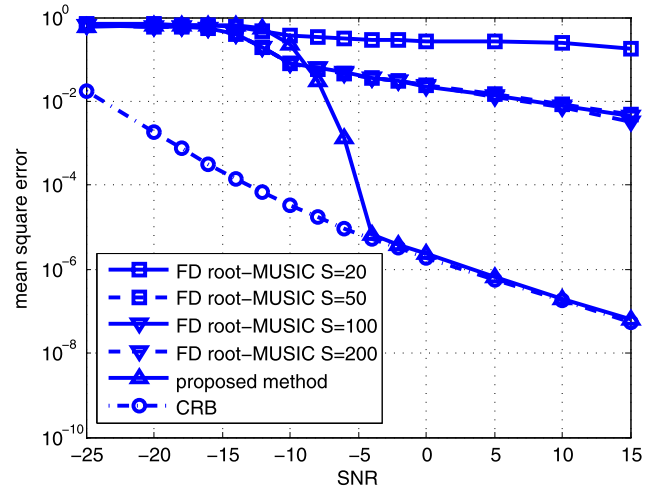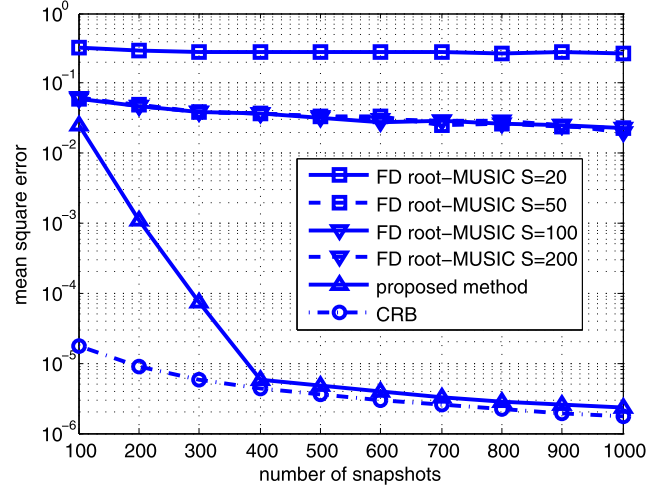
formance is much better than that of the FD root-MUSIC method with $S = 200$.

Moreover, in Figs. 6, 8, 10 and 12, we can observe the threshold effect that we mentioned in Section 3. For example, in Fig. 8, the MSE of the proposed method rapidly decreases around SNR $= -15$ dB; in Fig. 10 the threshold effect happens between the SNRs of -10 dB and -5 dB. These results agree with our discussion in Section 3.

In summary, we conclude that the proposed method compares favorably with the FD root-MUSIC method. It is important, however, that we also make a computational comparison between the methods. A drawback of the FD root-MUSIC method is that it requires the use of a large number of Fourier coefficients to achieve satisfactory performance. In [15] it was suggested that, in general, S, the number of Fourier coefficients, should be approximately equal to $4\kappa R$, where κ is the signal wavenumber, and R is the largest distance between the array sensors and the origin of the coordinate system. The complexity of FD root-MUSIC is $O((N + M)^2 K)$ [15]. Thus, to achieve satisfactory performance, the FD root-MUSIC method is computationally expensive even for arrays with small sizes. On the other hand, using the result of MODE in [17], we can obtain that the complexity of the proposed method is $O(M^2 K + N^2 K)$, which is smaller than that of FD root-MUSIC.
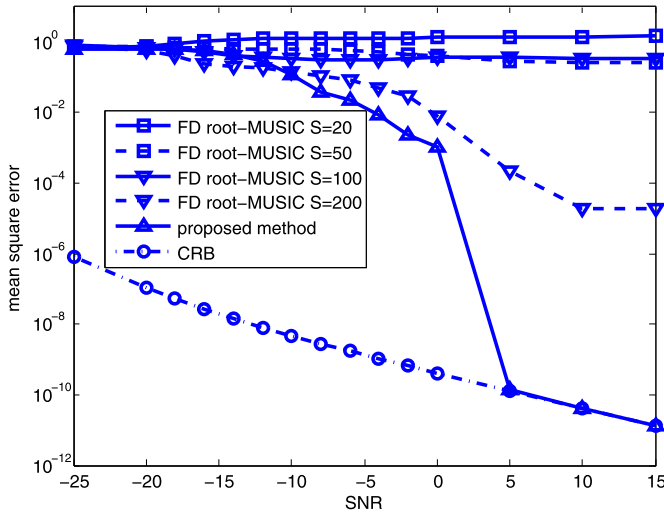
Fig. 12. Mean square error performance at different SNRs for three sources ($K = 100$). The coprime array is of size $(29, 31)$.
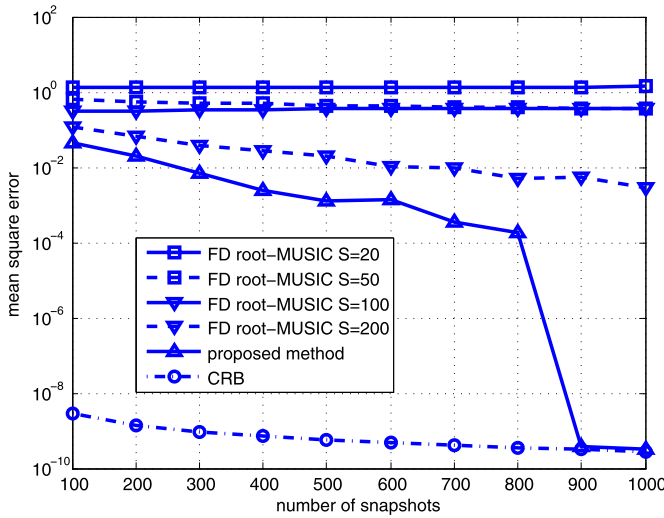

Fig. 13. Mean square error performance for different number of snapshots for three sources (SNR = -8 dB). The coprime array is of size $(29, 31)$.

5. Concluding remarks

In this paper, a fast search-free DOA estimation algorithm for coprime arrays is proposed. It exploits the uniform linear structure of the subarrays by first estimating the DOA separately within each subarray and then combining the estimates from different subarrays. Simulation shows that the performance of the proposed algorithm is close to the FD root-MUSIC method at low SNRs and much better at high SNRs. We point out that the proposed method has significantly lower complexity.

Here, we only discussed the case with two subarrays. For a coprime array with more than two subarrays, the state space will be a multi-dimensional hyperrectangle, but the entire angular domain can still be mapped to line segments and the same idea for esti-

mation can be reproduced. Therefore, the presented results can be extended to coprime arrays with multiple subarrays.

References

[1] P. Pal, P.P. Vaidyanathan, Coprime sampling and the MUSIC algorithm, in: Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE, IEEE, 2011, pp. 289–294.
[2] P.P. Vaidyanathan, P. Pal, Sparse sensing with co-prime samplers and arrays, IEEE Trans. Signal Process. 59 (2) (2011) 573–586.
[3] P.P. Vaidyanathan, P. Pal, Theory of sparse coprime sensing in multiple dimensions, IEEE Trans. Signal Process. 59 (8) (2011) 3592–3608.
[4] S. Negusse, P. Händel, The dual channel sinewave model: Co-prime sparse sampling, parameter estimation, and the Cramér–Rao bound, Measurement 45 (9) (2012) 2254–2263.
[5] J. Mayhan, L. Niro, Spatial spectral estimation using multiple beam antennas, IEEE Trans. Antennas Propag. 35 (8) (1987) 897–906.
[6] J. Capon, High-resolution frequency-wavenumber spectrum analysis, Proc. IEEE 57 (8) (1969) 1408–1418.
[7] R. Schmidt, Multiple emitter location and signal parameter estimation, IEEE Trans. Antennas Propag. 34 (3) (1986) 276–280.
[8] P. Stoica, A. Nehorai, Performance study of conditional and unconditional direction-of-arrival estimation, IEEE Trans. Acoust. Speech Signal Process. 38 (10) (1990) 1783–1795.
[9] Y. Bresler, A. Macovski, Exact maximum likelihood parameter estimation of superimposed exponential signals in noise, IEEE Trans. Acoust. Speech Signal Process. 34 (5) (1986) 1081–1089.
[10] P. Stoica, K.C. Sharman, Maximum likelihood methods for direction-of-arrival estimation, IEEE Trans. Acoust. Speech Signal Process. 38 (7) (1990) 1132–1143.
[11] B. Rao, K.V.S. Hari, Performance analysis of root-MUSIC, IEEE Trans. Acoust. Speech Signal Process. 37 (12) (1989) 1939–1949.
[12] R. Roy, T. Kailath, ESPRIT-estimation of signal parameters via rotational invariance techniques, IEEE Trans. Acoust. Speech Signal Process. 37 (7) (1989) 984–995.
[13] B. Friedlander, A.J. Weiss, Direction finding using spatial smoothing with interpolated arrays, IEEE Trans. Aerosp. Electron. Syst. 28 (2) (1992) 574–587.
[14] F. Belloni, A. Richter, V. Koivunen, DoA estimation via manifold separation for arbitrary array structures, IEEE Trans. Signal Process. 55 (10) (2007) 4800–4810.
[15] M. Rubsamen, A.B. Gershman, Direction-of-arrival estimation for nonuniform sensor arrays: from manifold separation to Fourier domain MUSIC methods, IEEE Trans. Signal Process. 57 (2) (2009) 588–599.
[16] H. Krim, M. Viberg, Two decades of array signal processing research: the parametric approach, IEEE Signal Process. Mag. 13 (4) (1996) 67–94.
[17] J. Li, P. Stoica, Z.S. Liu, Comparative study of IQML and MODE direction-of-arrival estimators, IEEE Trans. Signal Process. 46 (1) (1998) 149–160.
[18] H.L. Van Trees, Optimum Array Processing, Wiley–Interscience, 2002.
[19] A.B. Gershman, M. Rübsamen, M. Pesavento, One- and two-dimensional direction-of-arrival estimation: An overview of search-free techniques, Signal Process. 90 (5) (2010) 1338–1349.

Zhiyuan Weng was born in Shanghai, China. He is a Ph.D. candidate in the Department of Electrical and Computer Engineering at Stony Brook University. His research interests include statistical signal processing and machine learning.

Petar M. Djurić received his B.S. and M.S. degrees in electrical engineering from the University of Belgrade and his Ph.D. degree in electrical engineering from the University of Rhode Island. Currently, he is a professor in the Department of Electrical and Computer Engineering at Stony Brook University. He was an associate editor of several signal processing journals including IEEE Transactions on Signal Processing. In 2007, he received the IEEE Signal Processing Magazine Best Paper Award and, during 2008–2009, he was Distinguished Lecturer of the IEEE Signal Processing Society. In 2012, he received the European Association for Signal Processing Technical Achievement Award. He is a Fellow of the IEEE.