

Communication

# Fast Noncircular 2D-DOA Estimation for Rectangular Planar Array

Lingyun Xu <sup>1,2,\*</sup> and Fangqing Wen <sup>3</sup>

<sup>1</sup> College of Science, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China

<sup>2</sup> Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716, USA

<sup>3</sup> Electronic and Information School of Yangtze University, Jingzhou 434023, China; wfqzhc@163.com

\* Correspondence: xlyun@nuaa.edu.cn or xl\_yun@163.com; Tel.: +86-139-5203-5864

Academic Editors: Zhiguo Shi, Yujie Gu and Rongxing Lu

Received: 20 February 2017; Accepted: 31 March 2017; Published: 12 April 2017

**Abstract:** A novel scheme is proposed for direction finding with uniform rectangular planar array. First, the characteristics of noncircular signals and Euler's formula are exploited to construct a new real-valued rectangular array data. Then, the rotational invariance relations for real-valued signal space are depicted in a new way. Finally the real-valued propagator method is utilized to estimate the pairing two-dimensional direction of arrival (2D-DOA). The proposed algorithm provides better angle estimation performance and can discern more sources than the 2D propagator method. At the same time, it has very close angle estimation performance to the noncircular propagator method (NC-PM) with reduced computational complexity.

**Keywords:** two-dimensional direction of arrival (2D-DOA); real-valued propagator method; noncircular; rectangular planar array

## 1. Introduction

Two-dimensional direction of arrival (2D-DOA) estimation has been widely used in mobile communication systems, sonar, navigation, radar, etc. [1–5], which is an important research branch in array signal processing. Many 2D-DOA estimation algorithms have sprung up in recent years in order to improve the performance of angle estimation, which include the two dimensional multiple signal classification (2D MUSIC) algorithm [6], the 2D Unitary estimation of signal parameters via rotational invariance techniques (ESPRIT) algorithm [7], the modified 2D-ESPRIT algorithm [8], the matrix pencil method [9], the maximum likelihood method [10,11], the parallel factor (PARAFAC) algorithm [12], and so on [13–20]. However, those 2D-DOA estimation algorithms are confronted with the problem of the high computational complexity generally and they are very difficult to apply in engineering practice. As is known to us, the propagator method (PM) algorithm uses linear operations to replace the eigenvalue decomposition of the covariance matrix [21], and it has a great advantage in resolving the amount of calculation. Therefore, the 2D-DOA estimation based on PM is becoming a hot spot of research. For example, Wu et al. have developed the 2D-DOA estimation algorithm via the rotational invariance property of propagator matrix [22]. In [23], an improved PM algorithm is proposed for 2D-DOA estimation, which not only reduces the computational complexity, but also avoids the aperture loss.

Unfortunately, all the algorithms mentioned above did not consider the characteristics of the impinging signals. In fact, many noncircular signals such as the amplitude modulated (AM), binary phase shift keying (BPSK), minimum shift keying (MSK), and Gaussian MSK (GMSK) signals are used in wireless communication or satellite systems. In recent years, some scholars use non-circular signal characteristics to improve the performance of direction estimation, which contain the noncircular MUSIC (NC-MUSIC) algorithm [24], the NC-ESPRIT algorithm [25], and the noncircular parallel

factor (NC-PARAFAC) algorithm [26]. On the one hand, the angle estimation performance can be achieved by the algorithms [24–26]. On the other hand, the computation loads are increased greatly due to the doubled array aperture. The noncircular rational invariance propagator method has also been proposed for angle estimation in [27], which aimed at the linear array. If it is extended to the rectangular planar array for 2D-DOA estimation, the complexity would be increased greatly.

In this paper, we take advantage of the characteristics of noncircular signals and derive a novel noncircular propagator method algorithm based on the uniform rectangular planar array. The main works of this paper are listed in a straightforward manner as follows: (1) the property of the noncircular signal and Euler's transformation are used to construct a new real-valued rectangular array data; (2) the rotational invariance relations for real-valued signal space are depicted in a new way; (3) the PM algorithm is applied to two-dimensional angle estimation for the rectangular planar array which is paired automatically; and (4) theory analysis and simulation results confirm that our algorithm has better direction finding performance and can discern more sources than 2D-PM [23]. Due to real-valued processing, it can save about 75% computational load compared with the NC-PM algorithm [27]. However, its estimation performance is close to NC-PM algorithm, which has higher computational load.

## 2. Data Model

In order to get the two-dimensional direction finding, we consider a uniform rectangular planar array (URA) consisting of  $N$  uniform linear subarrays as shown in Figure 1, and there are  $M$  sensors in each subarray. The inter-element spacing between the two sensors is  $d$  in both the  $x$ -axis and  $y$ -axis. Suppose there are  $K$  narrowband far-field uncorrelated sources with wavelength  $\lambda$  impinging on the array from different directions. We also assume the noise is independent of the sources and  $d = \lambda/2$ . The output signal of the  $i$ th subarray  $\mathbf{x}_i(t)$  can be denoted as [26]:

$$\mathbf{x}_i(t) = \mathbf{A}_x \Phi^{i-1} \mathbf{S}(t) + \mathbf{n}_i(t), i = 1, 2, \dots, N, \quad (1)$$

where  $\mathbf{A}_x = [\mathbf{a}_x(v_1), \mathbf{a}_x(v_2), \dots, \mathbf{a}_x(v_K)]$  and  $\mathbf{a}_x(v_k) = [1, e^{-j\pi v_k}, \dots, e^{-j(M-1)\pi v_k}]^T$ ,  $v_k = \cos \phi_k \sin \theta_k$ ,  $\theta_k$  is the elevation angle and  $\phi_k$  is the azimuth angle.  $\Phi = \text{diag}(e^{-j\pi u_1}, e^{-j\pi u_2}, \dots, e^{-j\pi u_K})$  and  $u_k = \sin \phi_k \sin \theta_k$ .  $\mathbf{S}(t) = [\mathbf{s}_1(t), \mathbf{s}_2(t), \dots, \mathbf{s}_K(t)]^T$  is the noncircular signal vector. In addition, the vector of strictly second-order noncircular signals can be expressed as [28]:  $\mathbf{S}(t) = \mathbf{\Lambda} \mathbf{S}_o(t)$ ,  $\mathbf{S}_o(t) \in \mathbf{C}^{K \times 1}$ ,  $\mathbf{S}_o(t) = \mathbf{S}_o^*(t)$ , and  $\mathbf{\Lambda} = \text{diag}\{e^{j\varphi_1}, e^{j\varphi_2}, \dots, e^{j\varphi_K}\}$ , ( $e^{j\varphi_p} \neq e^{j\varphi_q}$ , for  $p \neq q$ ).  $\mathbf{n}_i(t)$  is the additive white Gaussian noise vector of the  $i$ th subarray.

Therefore, the whole array output is

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \\ \vdots \\ \mathbf{x}_N(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_x \\ \mathbf{A}_x \Phi \\ \vdots \\ \mathbf{A}_x \Phi^{N-1} \end{bmatrix} \mathbf{S}(t) + \begin{bmatrix} \mathbf{n}_1(t) \\ \mathbf{n}_2(t) \\ \vdots \\ \mathbf{n}_N(t) \end{bmatrix} = \mathbf{A} \mathbf{S}(t) + \mathbf{n}(t), \quad (2)$$

where  $\mathbf{A} = \mathbf{A}_y \circ \mathbf{A}_x$  is the  $MN \times K$  steering vector matrix,  $\circ$  represents the Khatri–Rao product, and  $\mathbf{A}_y = [\mathbf{a}_y(u_1), \mathbf{a}_y(u_2), \dots, \mathbf{a}_y(u_K)]$ ,  $\mathbf{a}_y(u_k) = [1, e^{-j\pi u_k}, \dots, e^{-j(N-1)\pi u_k}]^T$ , and  $\mathbf{n}(t) = [\mathbf{n}_1(t)^T, \mathbf{n}_2(t)^T, \dots, \mathbf{n}_N(t)^T]^T$ .

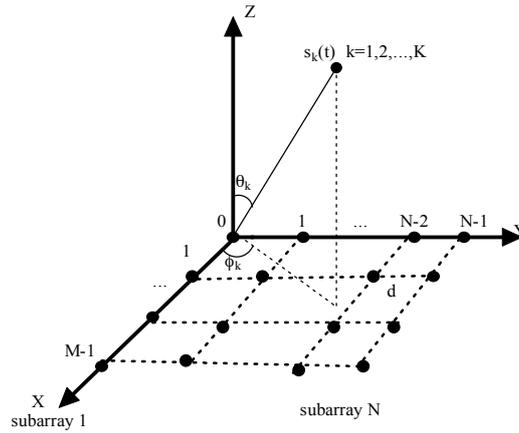


Figure 1. The structure of planar array.

### 3. Real-Valued PM Algorithm for 2D-DOA Estimation

#### 3.1. Euler Transformation

The real part and imaginary part of  $\mathbf{x}(t)$  can be obtained by utilizing the real-valued property of noncircular signals and Euler's formula as follows:

$$\mathbf{x}_R(t) = [\mathbf{x}(t) + \mathbf{x}^*(t)]/2 = \mathbf{A}_R \mathbf{S}_o(t) + \mathbf{n}_R(t), \quad (3)$$

$$\mathbf{x}_I(t) = [\mathbf{x}(t) - \mathbf{x}^*(t)]/(-2j) = \mathbf{A}_I \mathbf{S}_o(t) + \mathbf{n}_I(t), \quad (4)$$

where  $\mathbf{n}_R(t) = [\mathbf{n}(t) + \mathbf{n}^*(t)]/2$ ,  $\mathbf{n}_I(t) = [\mathbf{n}(t) - \mathbf{n}^*(t)]/(-2j)$ ,

$$\mathbf{A}_R = \begin{bmatrix} \left. \begin{array}{cccc} \cos \varphi_1 & \cos \varphi_2 & \cdots & \cos \varphi_K \\ \cos(\pi v_1 + \varphi_1) & \cos(\pi v_2 + \varphi_2) & \cdots & \cos(\pi v_K + \varphi_K) \\ \vdots & \vdots & \vdots & \vdots \\ \cos[\pi(M-1)v_1 + \varphi_1] & \cos[\pi(M-1)v_2 + \varphi_2] & \cdots & \cos[\pi(M-1)v_K + \varphi_K] \end{array} \right\} M \\ \left. \begin{array}{cccc} \cos(\pi u_1 + \varphi_1) & \cos(\pi u_2 + \varphi_2) & \cdots & \cos(\pi u_K + \varphi_K) \\ \cos[\pi(v_1 + u_1) + \varphi_1] & \cos[\pi(v_2 + u_2) + \varphi_2] & \cdots & \cos[\pi(v_K + u_K) + \varphi_K] \\ \vdots & \vdots & \vdots & \vdots \\ \cos[\pi((M-1)v_1 + u_1) + \varphi_1] & \cos[\pi((M-1)v_2 + u_2) + \varphi_2] & \cdots & \cos[\pi((M-1)v_K + u_K) + \varphi_K] \end{array} \right\} M \\ \left. \begin{array}{cccc} \cos[\pi(N-1)u_1 + \varphi_1] & \cos[\pi(N-1)u_2 + \varphi_2] & \cdots & \cos[\pi(N-1)u_K + \varphi_K] \\ \cos[\pi(N-1)u_1 + \pi v_1 + \varphi_1] & \cos[\pi(N-1)u_2 + \pi v_2 + \varphi_2] & \cdots & \cos[\pi(N-1)u_K + \pi v_K + \varphi_K] \\ \vdots & \vdots & \vdots & \vdots \\ \cos[\pi(N-1)u_1 + \pi(M-1)v_1 + \varphi_1] & \cos[\pi(N-1)u_2 + \pi(M-1)v_2 + \varphi_2] & \cdots & \cos[\pi(N-1)u_K + \pi(M-1)v_K + \varphi_K] \end{array} \right\} M \end{bmatrix} \in \mathbf{R}^{MN \times K}$$

$$\mathbf{A}_I = \begin{bmatrix} \left. \begin{array}{cccc} \sin \varphi_1 & \sin \varphi_2 & \cdots & \sin \varphi_K \\ \sin(\pi v_1 + \varphi_1) & \sin(\pi v_2 + \varphi_2) & \cdots & \sin(\pi v_K + \varphi_K) \\ \vdots & \vdots & \vdots & \vdots \\ \sin[\pi(M-1)v_1 + \varphi_1] & \sin[\pi(M-1)v_2 + \varphi_2] & \cdots & \sin[\pi(M-1)v_K + \varphi_K] \end{array} \right\} M \\ \left. \begin{array}{cccc} \sin(\pi u_1 + \varphi_1) & \sin(\pi u_2 + \varphi_2) & \cdots & \sin(\pi u_K + \varphi_K) \\ \sin[\pi(v_1 + u_1) + \varphi_1] & \sin[\pi(v_2 + u_2) + \varphi_2] & \cdots & \sin[\pi(v_K + u_K) + \varphi_K] \\ \vdots & \vdots & \vdots & \vdots \\ \sin[\pi((M-1)v_1 + u_1) + \varphi_1] & \sin[\pi((M-1)v_2 + u_2) + \varphi_2] & \cdots & \sin[\pi((M-1)v_K + u_K) + \varphi_K] \end{array} \right\} M \\ \left. \begin{array}{cccc} \sin[\pi(N-1)u_1 + \varphi_1] & \sin[\pi(N-1)u_2 + \varphi_2] & \cdots & \sin[\pi(N-1)u_K + \varphi_K] \\ \sin[\pi(N-1)u_1 + \pi v_1 + \varphi_1] & \sin[\pi(N-1)u_2 + \pi v_2 + \varphi_2] & \cdots & \sin[\pi(N-1)u_K + \pi v_K + \varphi_K] \\ \vdots & \vdots & \vdots & \vdots \\ \sin[\pi(N-1)u_1 + \pi(M-1)v_1 + \varphi_1] & \sin[\pi(N-1)u_2 + \pi(M-1)v_2 + \varphi_2] & \cdots & \sin[\pi(N-1)u_K + \pi(M-1)v_K + \varphi_K] \end{array} \right\} M \end{bmatrix} \in \mathbf{R}^{MN \times K}$$

Then, we define a new virtual array data as follows:

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{x}_R(t) \\ \mathbf{x}_I(t) \end{bmatrix} = \mathbf{B}\mathbf{S}_o(t) + \mathbf{n}_r(t), \tag{5}$$

where  $\mathbf{B} = \begin{bmatrix} \mathbf{A}_R \\ \mathbf{A}_I \end{bmatrix} \in \mathbf{R}^{2MN \times K}$ ,  $\mathbf{n}_r(t) = \begin{bmatrix} \mathbf{n}_R(t) \\ \mathbf{n}_I(t) \end{bmatrix} \in \mathbf{R}^{2MN \times 1}$ .

Define two matrices as follows:  $\mathbf{T}_1 = [\mathbf{I}_{M(N-1)} \mathbf{0}_{M(N-1) \times M}] \in \mathbf{R}^{M(N-1) \times MN}$ ,  $\mathbf{T}_2 = [\mathbf{0}_{M(N-1) \times M} \mathbf{I}_{M(N-1)}] \in \mathbf{R}^{M(N-1) \times MN}$ ; then, we construct two matrices  $\mathbf{J}_1 = [\mathbf{T}_1 + \mathbf{T}_2, \mathbf{0}_{M(N-1) \times MN}] \in \mathbf{R}^{M(N-1) \times 2MN}$  and  $\mathbf{J}_2 = [\mathbf{0}_{M(N-1) \times MN}, \mathbf{T}_2 - \mathbf{T}_1] \in \mathbf{R}^{M(N-1) \times 2MN}$ , and we can get the following relationship:

$$\mathbf{J}_2 \mathbf{B} = \mathbf{J}_1 \mathbf{B} \mathbf{G}, \tag{6}$$

where  $\mathbf{G} = \text{diag}\{\tan(\frac{\pi u_1}{2}), \tan(\frac{\pi u_2}{2}), \dots, \tan(\frac{\pi u_K}{2})\}$  is a real-valued matrix whose diagonal elements contain the needed angle information:

$$\mathbf{J}_1 \mathbf{B} = \begin{bmatrix} \left. \begin{array}{ccc} 2 \cos \frac{\pi u_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{\pi u_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi u_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ 2 \cos \frac{2\pi v_1 + \pi u_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{2\pi v_2 + \pi u_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{2\pi v_K + \pi u_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{2\pi(M-1)v_1 + \pi u_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{2\pi(M-1)v_2 + \pi u_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{2\pi(M-1)v_K + \pi u_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ 2 \cos \frac{3\pi u_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ 2 \cos \frac{3\pi u_1 + 2\pi v_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\pi v_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\pi v_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{3\pi u_1 + 2\pi(M-1)v_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\pi(M-1)v_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\pi(M-1)v_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \end{array} \right\} M \\ \left. \begin{array}{ccc} \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\pi v_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\pi v_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\pi v_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\pi(M-1)v_1 + 2\phi_1}{2} \cos \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\pi(M-1)v_2 + 2\phi_2}{2} \cos \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\pi(M-1)v_K + 2\phi_K}{2} \cos \frac{\pi u_K}{2} \end{array} \right\} M \end{bmatrix} \in \mathbf{R}^{M(N-1) \times K}$$

$$\mathbf{J}_2 \mathbf{B} = \begin{bmatrix} \left. \begin{array}{ccc} 2 \cos \frac{\pi u_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{\pi u_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi u_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ 2 \cos \frac{2\pi v_1 + \pi u_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{2\pi v_2 + \pi u_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{2\pi v_K + \pi u_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{2\pi(M-1)v_1 + \pi u_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{2\pi(M-1)v_2 + \pi u_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{2\pi(M-1)v_K + \pi u_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ 2 \cos \frac{3\pi u_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ 2 \cos \frac{3\pi u_1 + 2\pi v_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\pi v_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\pi v_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{3\pi u_1 + 2\pi(M-1)v_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{3\pi u_2 + 2\pi(M-1)v_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{3\pi u_K + 2\pi(M-1)v_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \end{array} \right\} M \\ \left. \begin{array}{ccc} \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\pi v_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\pi v_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\pi v_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \\ \vdots & \vdots & \vdots & \vdots \\ 2 \cos \frac{\pi(2N-3)u_1 + 2\pi(M-1)v_1 + 2\phi_1}{2} \sin \frac{\pi u_1}{2} & 2 \cos \frac{\pi(2N-3)u_2 + 2\pi(M-1)v_2 + 2\phi_2}{2} \sin \frac{\pi u_2}{2} & \dots & 2 \cos \frac{\pi(2N-3)u_K + 2\pi(M-1)v_K + 2\phi_K}{2} \sin \frac{\pi u_K}{2} \end{array} \right\} M \end{bmatrix} \in \mathbf{R}^{M(N-1) \times K}$$

Similarly, define two  $(M - 1) \times M$  Toeplitz matrices as follows:  $\mathbf{T}_3 = \begin{bmatrix} 1 & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 1 & 1 \end{bmatrix}$ ,

$$\mathbf{T}_4 = \begin{bmatrix} 1 & -1 & 0 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 1 & -1 \end{bmatrix}.$$

Then, we construct two matrices  $\mathbf{J}_3$  and  $\mathbf{J}_4$  as follows:  $\mathbf{J}_3 = \mathbf{I}_{2N} \otimes \mathbf{T}_3 \in \mathbf{R}^{2(M-1)N \times 2MN}$  and

$$\mathbf{J}_4 = \begin{bmatrix} \mathbf{0} & -\mathbf{I}_N \otimes \mathbf{T}_4 \\ \mathbf{I}_N \otimes \mathbf{T}_4 & \mathbf{0} \end{bmatrix} \in \mathbf{R}^{2(M-1)N \times 2MN}.$$

We also get the following relationship:

$$\mathbf{J}_4 \mathbf{B} = \mathbf{J}_3 \mathbf{B} \mathbf{D}, \quad (7)$$

where  $\mathbf{D} = \text{diag}\{\tan(\frac{\pi v_1}{2}), \tan(\frac{\pi v_2}{2}), \dots, \tan(\frac{\pi v_K}{2})\}$  is a real-valued matrix whose diagonal elements also contain the desired angle information.

### 3.2. 2D-DOA Estimation

According to Equation (5), the estimation of covariance matrix  $\mathbf{R}$  of  $\mathbf{y}(t)$  is denoted by collecting  $L$  snapshots:

$$\mathbf{R} = \frac{1}{L} \sum_{i=1}^L \mathbf{y}(t_i) \mathbf{y}^H(t_i). \quad (8)$$

From Equation (8),  $\mathbf{R}$  can be denoted by  $\mathbf{R} = \begin{bmatrix} \mathbf{R}_{x1} \\ \mathbf{R}_{x2} \end{bmatrix}$ , where  $\mathbf{R}_{x1} \in \mathbf{R}^{K \times 2MN}$ ,  $\mathbf{R}_{x2} \in \mathbf{R}^{(2MN-K) \times 2MN}$ . In the noiseless case,  $\mathbf{R}_{x1} \mathbf{p}_r = \mathbf{R}_{x2}$ , an estimation matrix  $\mathbf{p}_r$  can be obtained by [21]:

$$\hat{\mathbf{p}}_r = (\mathbf{R}_{x1} \mathbf{R}_{x1}^H)^{-1} \mathbf{R}_{x1} \mathbf{R}_{x2}^H. \quad (9)$$

We construct a new matrix  $\mathbf{p}_s = \begin{bmatrix} \mathbf{I}_K \\ \hat{\mathbf{p}}_r^H \end{bmatrix}$ , where  $\mathbf{I}_K$  is the identity matrix. In the noiseless case, the relationship between  $\mathbf{p}_s$  and  $\mathbf{B}$  can be obtained by a unique non-singular matrix  $\mathbf{T}$  as

$$\mathbf{p}_s = \mathbf{B} \mathbf{T}. \quad (10)$$

Substituting Equation (10) into Equation (6), we can get

$$\mathbf{J}_2 \mathbf{p}_s \mathbf{T}^{-1} = \mathbf{J}_1 \mathbf{p}_s \mathbf{T}^{-1} \mathbf{G}. \quad (11)$$

If we define  $\mathbf{P}_1 = (\mathbf{J}_1 \mathbf{p}_s)^\dagger \mathbf{J}_2 \mathbf{p}_s$ , we then have

$$\mathbf{P}_1 = \mathbf{T}^{-1} \mathbf{G} \mathbf{T}. \quad (12)$$

Equation (12) shows that the diagonal elements of the matrix  $\mathbf{G}$  can be obtained by performing the eigenvalue decomposition of  $\mathbf{P}_1$ , and  $\mathbf{T}$  is the corresponding eigenvector.

Then, we can get the estimation of  $\hat{u}_k$ :

$$\hat{u}_k = 2 \arctan(\hat{\rho}_k) / \pi, \quad (13)$$

where  $\hat{\rho}_k$  is the  $k$ th diagonal element of the matrix  $\mathbf{G}$ .

Similarly, Substituting Equation (10) into Equation (7), we can also get

$$\mathbf{J}_4 \mathbf{p}_s \mathbf{T}^{-1} = \mathbf{J}_3 \mathbf{p}_s \mathbf{T}^{-1} \mathbf{D}. \quad (14)$$

If we define  $\mathbf{P}_2 = (\mathbf{J}_4 \mathbf{p}_s)^\dagger \mathbf{J}_3 \mathbf{p}_s$ , we then have

$$\mathbf{P}_2 = \mathbf{T}^{-1} \mathbf{D} \mathbf{T}. \quad (15)$$

Then, we get the estimation of  $\hat{v}_k$ :

$$\hat{v}_k = 2 \arctan(\hat{\rho}_k) / \pi, \quad (16)$$

where  $\hat{\rho}_k$  is the  $k$ th diagonal element of the matrix  $\mathbf{D}$ .

We note that  $\hat{u}_k$  and  $\hat{v}_k$  share the same eigenvector  $\mathbf{T}$ , so the pairing is automatically formed. Thus, 2D-DOA can be obtained by

$$\hat{\theta}_k = \sin^{-1} \left( \sqrt{\hat{u}_k^2 + \hat{v}_k^2} \right), \quad (17)$$

$$\hat{\phi}_k = \tan^{-1}(\hat{u}_k / \hat{v}_k). \quad (18)$$

We have now achieved the essence of the proposed algorithm. The major algorithmic steps are as follows:

- (1) Construct the matrix  $\mathbf{y}(t)$  from Equation (5), and compute the covariance matrix  $\mathbf{R}$  of  $\mathbf{y}(t)$  through Equation (8).
- (2) Estimation of the propagator  $\mathbf{p}_r$  from Equation (9), and then construct the matrix  $\mathbf{p}_s$ .
- (3) Construct the matrix  $\mathbf{J}_1\mathbf{p}_s$  and  $\mathbf{J}_2\mathbf{p}_s$  and perform the eigenvalue decomposition of  $\mathbf{P}_1 = (\mathbf{J}_1\mathbf{p}_s)^\dagger \mathbf{J}_2\mathbf{p}_s$ .
- (4) Similarly, construct the matrix  $\mathbf{J}_3\mathbf{p}_s$  and  $\mathbf{J}_4\mathbf{p}_s$  and perform the eigenvalue decomposition of  $\mathbf{P}_2 = (\mathbf{J}_3\mathbf{p}_s)^\dagger \tilde{\mathbf{J}}_4\mathbf{p}_s$ .
- (5) Finally, estimate the 2D-DOA through Equations (17) and (18).

**Remark 1.** In [23], the conventional PM algorithm divides the steering matrix  $\mathbf{A}$  into two matrices  $\mathbf{A}_1 \in \mathbf{C}^{K \times K}$  and  $\mathbf{A}_2 \in \mathbf{C}^{(MN-K) \times K}$ , and  $\mathbf{A}_2$  is the linear transformation of  $\mathbf{A}_1$ , i.e.,  $\mathbf{A}_2 = \mathbf{P}_m^H \mathbf{A}_1$ ,  $\mathbf{P}_m \in \mathbf{C}^{K \times (MN-K)}$  is the propagator operator. According to Equation (1),  $\mathbf{x}(t) = \mathbf{A}\mathbf{S}(t) + \mathbf{n}(t)$ , and the covariance matrix of received data  $\mathbf{x}(t) \in \mathbf{C}^{MN \times 1}$  is  $\mathbf{R}_p = \frac{1}{L} \sum_{i=1}^L \mathbf{x}(t_i) \mathbf{x}^H(t_i)$ . We partition it as  $\mathbf{R}_p = \begin{bmatrix} \mathbf{R}_{p1} \\ \mathbf{R}_{p2} \end{bmatrix}$ , where  $\mathbf{R}_{p1} \in \mathbf{R}^{K \times MN}$ ,  $\mathbf{R}_{p2} \in \mathbf{R}^{(MN-K) \times MN}$ , and we can get the propagator estimator  $\hat{\mathbf{p}}_m = (\mathbf{R}_{p1} \mathbf{R}_{p1}^H)^{-1} \mathbf{R}_{p1} \mathbf{R}_{p2}^H$ . In our paper, according to Equation (5),  $\mathbf{y}(t) = \begin{bmatrix} \mathbf{x}_R(t) \\ \mathbf{x}_I(t) \end{bmatrix} = \mathbf{B}\mathbf{S}_o(t) + \mathbf{n}_r(t)$ , and we compute the covariance of  $\mathbf{y}(t) \in \mathbf{R}^{2MN \times 1}$  to estimate the propagator. Apparently, the available array aperture of the proposed algorithm can be thought of as twice that of the conventional 2D-PM [23], so it has better angle performance than 2D-PM.

**Remark 2.** In [23], define  $\mathbf{p}_c = \begin{bmatrix} \mathbf{I}_K \\ \hat{\mathbf{P}}_m^H \end{bmatrix}$ , and then  $\mathbf{p}_c \mathbf{A}_1 = \mathbf{A}$ , which means that the columns in  $\mathbf{p}_c$  span the same signal subspace as the column vectors in  $\mathbf{A}$ . Divide  $\mathbf{p}_c$  into  $\mathbf{p}_{c1} \in \mathbf{C}^{M(N-1) \times K}$  and  $\mathbf{p}_{c2} \in \mathbf{C}^{M(N-1) \times K}$ ,  $\mathbf{p}_{c1}$ ,  $\mathbf{p}_{c2}$  are the first  $M(N-1)$  rows and the last  $M(N-1)$  rows of  $\mathbf{p}_c$ . Then, get the relationship,  $\mathbf{P}_{c1}^+ \mathbf{P}_{c2} = \mathbf{A}_1 \Phi \mathbf{A}_1$ , where  $\Phi = \text{diag}(e^{-j\pi u_1}, e^{-j\pi u_2}, \dots, e^{-j\pi u_K})$ . Perform the eigenvalue decomposition of  $\mathbf{P}_{c1}^+ \mathbf{P}_{c2}$  to obtain the diagonal elements of the matrix  $\Phi$ . Similarly, reconstruct  $\mathbf{p}_c$  to  $\mathbf{p}'_c$ ,  $\mathbf{p}'_{c1}$ ,  $\mathbf{p}'_{c2}$  being the first  $N(M-1)$  rows and the last  $N(M-1)$  rows of  $\mathbf{p}'_c$ , and perform the eigenvalue decomposition of  $\mathbf{P}'_{c1} + \mathbf{P}'_{c2}$  to obtain the diagonal elements of the matrix  $\Phi_x$ , where  $\Phi_x = \text{diag}(e^{-j\pi v_1}, e^{-j\pi v_2}, \dots, e^{-j\pi v_K})$ . Finally, the 2D-DOA can be obtained from the diagonal elements of  $\Phi$  and  $\Phi_x$ . From the above mentioned, the row dimensions of  $\mathbf{p}_{c1}$ ,  $\mathbf{p}_{c2}$  and  $\mathbf{p}'_{c1}$ ,  $\mathbf{p}'_{c2}$  are equal to  $M(N-1)$ ,  $(M-1)N$ , respectively. The maximum number of the identified sources is  $\min[M(N-1), (M-1)N]$ . In our proposed algorithm, from Equation (11) and Equation (14), the row dimensions of  $\mathbf{J}_1\mathbf{p}_s$  and  $\mathbf{J}_2\mathbf{p}_s$ ,  $\mathbf{J}_3\mathbf{p}_s$  and  $\mathbf{J}_4\mathbf{p}_s$  are equal to  $M(N-1)$ ,  $2(M-1)N$ , respectively. Therefore, the maximum number of the identified sources is  $\min[M(N-1), 2(M-1)N]$ . If  $M < N$ , the proposed algorithm can discern more sources than that of the conventional 2D-PM [23].

**Remark 3.** In the NC-PM algorithm [27], according to Equation (2), the extended array output data is denoted as  $\mathbf{Y} = \begin{bmatrix} \mathbf{X} \\ \mathbf{J}_{MN} \mathbf{X}^* \end{bmatrix} = \mathbf{A}_{nc} \mathbf{S}_o + \mathbf{N}_{nc}$ , where  $\mathbf{A}_{nc} \in \mathbf{C}^{2MN \times K}$ ,  $\mathbf{J}_{MN}$  is the  $MN \times MN$  exchange matrix with ones on its anti-diagonal and zeros elsewhere, and  $\mathbf{X}^* \in \mathbf{C}^{MN \times L}$  stands for the complex conjugation of  $\mathbf{X}$ ,  $\mathbf{Y} \in \mathbf{C}^{2MN \times L}$ . Compute the covariance of  $\mathbf{Y}$  to estimate the propagator  $\mathbf{p}_{nc}$ . Similarly, the invariance

equations for  $\mathbf{p}_{nc}$  are constructed to estimate the 2D-DOA. As is known to us, each computation amount of the complex multiplication is four times that of the real-valued one. In our algorithm, we use Euler transformation to convert complex arithmetic of noncircular to real arithmetic. For example, according to Equation (5),  $\mathbf{y}(t) = \mathbf{B}\mathbf{S}_o(t) + \mathbf{n}_r(t)$ ,  $\mathbf{y}(t) \in \mathbf{R}^{2MN \times 1}$ , and the computation amounts of covariance of  $\mathbf{y}(t)$  with snapshots  $L$  are much lower than that of  $\mathbf{Y}$  [27]. Due to real-valued processing, our algorithm can save about 75% computational load compared with the NC-PM algorithm [27].

#### 4. Cramer-Rao Bounds and Analysis

##### 4.1. CRB

In this section, we give the Cramer-Rao Bounds (CRB) of noncircular signal for rectangular planar array. According to Equation (5), the received data is

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{x}_R(t) \\ \mathbf{x}_I(t) \end{bmatrix} = \mathbf{B}\mathbf{S}_o(t) + \mathbf{n}_r(t), \quad (19)$$

where  $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_K] \in \mathbf{C}^{2MN \times K}$ , and  $\mathbf{n}_r(t)$  is the noise vector. The Fisher information matrix (FIM) in relation to  $\phi = [\phi_1, \phi_2, \dots, \phi_K]$  and  $\theta = [\theta_1, \theta_2, \dots, \theta_K]$  can be calculated as follows [29]:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix}. \quad (20)$$

According to [29], we know that the  $(i, j)$  ith element of  $\mathbf{F}_{11}$  is given by

$$\begin{aligned} \mathbf{F}(\theta_i, \theta_j) &= 2\text{Re}[\text{trace}(\dot{\mathbf{B}}_{\theta_i} \mathbf{S}_o)^H \Gamma^{-1}(\dot{\mathbf{B}}_{\theta_j} \mathbf{S}_o)] \\ &= 2\text{Re}[\text{trace}(\dot{\mathbf{B}}_{\theta_i} \mathbf{e}_i \mathbf{e}_i^T \mathbf{S}_o)^H \Gamma^{-1}(\dot{\mathbf{B}}_{\theta_j} \mathbf{e}_j \mathbf{e}_j^T \mathbf{S}_o)] \\ &= \text{Re}[\text{trace}(\mathbf{S}_o^H \mathbf{e}_i \mathbf{e}_i^T \dot{\mathbf{B}}_{\theta_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\theta_j} \mathbf{e}_j \mathbf{e}_j^T \mathbf{S}_o)] \\ &= \text{Re}[\text{trace}(\mathbf{e}_i^T \dot{\mathbf{B}}_{\theta_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\theta_j} \mathbf{e}_j) (\mathbf{e}_j^T \mathbf{S}_o \mathbf{S}_o^H \mathbf{e}_i)] \\ &= 2L\text{Re}[(\dot{\mathbf{B}}_{\theta_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\theta_j})_{ij} (\mathbf{R}_{s_o}^T)_{ij}] \end{aligned} \quad (21)$$

Likely, we can give the  $(i, j)$  ith element of  $\mathbf{F}_{12}$ ,  $\mathbf{F}_{21}$ ,  $\mathbf{F}_{22}$ :

$$\mathbf{F}(\theta_i, \phi_j) = 2L\text{Re}[(\dot{\mathbf{B}}_{\theta_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\phi_j})_{ij} (\mathbf{R}_{s_o}^T)_{ij}], \quad (22)$$

$$\mathbf{F}(\phi_i, \theta_j) = 2L\text{Re}[(\dot{\mathbf{B}}_{\phi_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\theta_j})_{ij} (\mathbf{R}_{s_o}^T)_{ij}], \quad (23)$$

$$\mathbf{F}(\phi_i, \phi_j) = 2L\text{Re}[(\dot{\mathbf{B}}_{\phi_i}^H \Gamma^{-1} \dot{\mathbf{B}}_{\phi_j})_{ij} (\mathbf{R}_{s_o}^T)_{ij}], \quad (24)$$

where  $\mathbf{e}_i$  denotes the  $i$ th column of the unit matrix,  $\mathbf{R}_{s_o} = \frac{1}{L} \mathbf{S}_o \mathbf{S}_o^H$ ,  $\dot{\mathbf{B}}_{\zeta_i} = \frac{\partial \mathbf{B}}{\partial \zeta_i}$ ,  $\dot{\mathbf{B}}_{\zeta} = [\frac{\partial \mathbf{B}}{\partial \zeta_1}, \frac{\partial \mathbf{B}}{\partial \zeta_2}, \dots, \frac{\partial \mathbf{B}}{\partial \zeta_K}]$ ,

$$\mathbf{\Gamma} = \begin{bmatrix} \sigma^2_{2MN} & 0 & \dots & 0 \\ 0 & \sigma^2_{2MN} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma^2_{2MN} \end{bmatrix}, \quad \sigma^2 \text{ is the covariance of the noise.} \quad \text{According to}$$

Equations (21)–(24), we can obtain:

$$\mathbf{F}_{11} = 2L\text{Re}[(\dot{\mathbf{B}}_{\theta}^H \Gamma^{-1} \dot{\mathbf{B}}_{\theta}) \oplus (\mathbf{R}_{s_o}^T)], \quad (25)$$

$$\mathbf{F}_{12} = 2L\text{Re}[(\dot{\mathbf{B}}_{\theta}^H \Gamma^{-1} \dot{\mathbf{B}}_{\phi}) \oplus (\mathbf{R}_{s_o}^T)], \quad (26)$$

$$\mathbf{F}_{21} = 2L\text{Re}[(\dot{\mathbf{B}}_\phi^H \Gamma^{-1} \dot{\mathbf{B}}_\theta) \oplus (\mathbf{R}_{s_0}^T)], \quad (27)$$

$$\mathbf{F}_{22} = 2L\text{Re}[(\dot{\mathbf{B}}_\phi^H \Gamma^{-1} \dot{\mathbf{B}}_\phi) \oplus (\mathbf{R}_{s_0}^T)], \quad (28)$$

where  $\oplus$  represents Hadamard product.

Then, the CRB can be denoted as:

$$\text{CRB} = \mathbf{F}^{-1}. \quad (29)$$

We present the curves of CRB versus different signal to noise ratios (SNRs) and snapshots  $L$  in Figures 2 and 3. The source number  $K$  is fixed at 3  $M$  and  $N$  represents the numbers of sensors on the  $x$ -axis and the  $y$ -axis. In Figure 2, the snapshot  $L$  is fixed at 200. It is obvious that, with the improvement of SNR, the value of CRB decreases accordingly. In Figure 3, we set SNR at 20 dB, and the curve shows that the value of CRB decreases with increase of  $L$ , and simulation results and theory analysis are consistent.

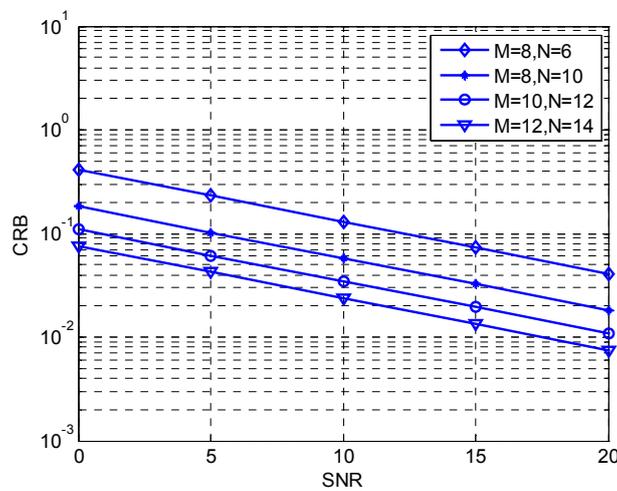


Figure 2. CRB comparison versus SNR.

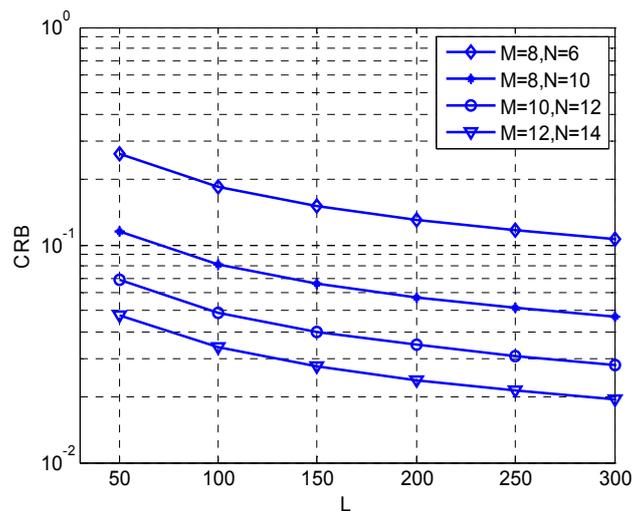


Figure 3. CRB comparison versus snapshots  $L$ .

#### 4.2. Complexity Analysis

In this section, we analyse the computational complexity of the algorithm specifically. First, estimation of the covariance matrix  $\mathbf{R}$  requires  $O(4M^2N^2L)$  real-valued multiplications

(RMS). In addition, the estimation of the matrix  $\mathbf{p}_s$  takes  $O(2MNK^2 + 4M^2N^2K + K^3)$  RMS. Then, the estimation and eigenvalue decomposition of the matrix  $\mathbf{P}_1$  and  $\mathbf{P}_2$  totally require  $O(4M^2NK(N-1) + 3M(N-1)K^2 + 8M(M-1)N^2K + 2K^3)$  RMS. Therefore, the overall computational complexity of our algorithm is  $O(4M^2N^2L + 5MNK^2 + 16M^2N^2K + 3K^3 - 4M^2NK - 3MK^2 - 8MN^2K)$  RMS. As we know that each computation amount of the complex multiplication is four times that of the real-valued one, we can show the Chen's noncircular propagator algorithm [27] needs  $O(16M^2N^2L + 8MNK^2 + 16M^2N^2K + 3K^3 + 16M(N-1)K^2)$  RMS, J's noncircular ESPRIT [25] needs  $O(16M^2N^2L + M^3N^3 + 8M(N-1)K^2 + 3K^3 + 8N(M-1)K^2)$  RMS, Zhang's 2D-ESPRIT algorithm [8] needs  $O(4M^2N^2L + M^3N^3 + 4M(N-1)K^2 + 4N(M-1)K^2 + 6K^3)$  RMS, while Li's 2D-PM [23] requires  $O(4M^2N^2L + 4MNK^2 + 4M^2N^2K + 6K^3 + 4M(N-1)K^2 + 4N(M-1)K^2)$  RMS.

The complexity comparisons with different parameters are shown in Figures 4 and 5. In Figure 4, the numbers of sensor  $M$  and  $N$  on the  $x$ -axis and the  $y$ -axis are set at 8 and 6, respectively. The source number  $K$  is fixed at 3. In Figure 5, the parameters  $N$  and  $K$  are the same as Figure 4, and the snapshot  $L$  is set to 100. From Figures 4 and 5, we can observe that the proposed algorithm has much lower computational load than J's NC-ESPRIT algorithm and Chen's NC-PM algorithm.

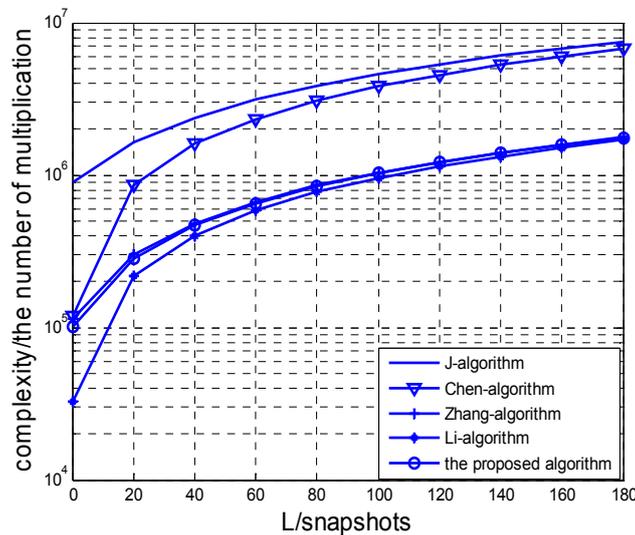


Figure 4. Complexity comparison versus  $L$ .

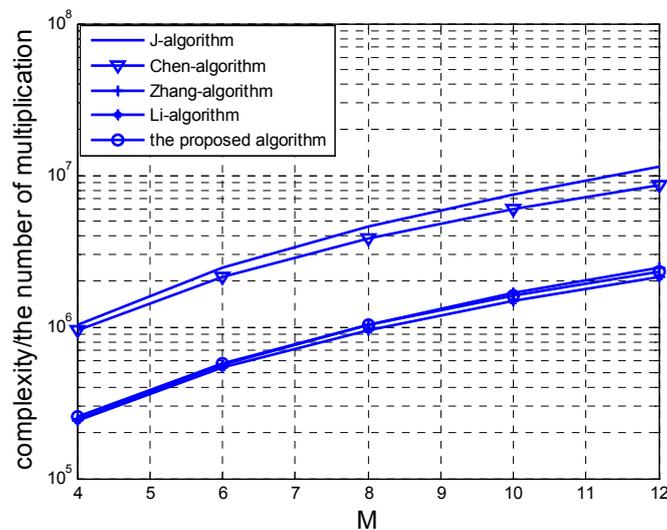


Figure 5. Complexity comparison versus  $M$ .

We can summarize the merits of the proposed algorithm as follows:

- (1) The proposed algorithm has much lower computational load than the NC-PM and NC-ESPRIT algorithms because the proposed algorithm uses Euler transformation to convert complex arithmetic of noncircular PM to real arithmetic.
- (2) The proposed algorithm has better estimation performance than the 2D-PM algorithm because the array aperture is doubled according to Equation (5).
- (3) The maximum number of discerned sources of our algorithm is dependent on Equation (5) and the real-valued PM method. Obviously, the maximum number of the identified sources of our proposed algorithm is  $\min[M(N-1), 2(M-1)N]$ , while 2D-PM is  $\min[M(N-1), (M-1)N]$ .
- (4) The proposed algorithm requires no extra matching calculation. The estimated 2D-DOA can automatically be matched.

## 5. Simulation Results

In this section, we use Monte Carlo simulations to verify the performance of the algorithm. In the simulation, the rectangular planar array is configured with  $N$  subarrays, each subarray contains  $M$  sensors,  $L$  is the snapshots of the sources, and  $K$  is the number of the sources. We assume that there are  $K = 3$  non-coherent sources, which are BPSK modulated in Figures 4–8, where  $(\theta_1, \phi_1) = (15^\circ, 10^\circ)$ ,  $(\theta_2, \phi_2) = (25^\circ, 20^\circ)$  and  $(\theta_3, \phi_3) = (35^\circ, 30^\circ)$ , respectively.

The root mean squared error (RMSE) is used for performance assessment, which is defined as  $\frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{1000} \sum_{n=1}^{1000} (\hat{\theta}_{k,n} - \theta_k)^2 + (\hat{\phi}_{k,n} - \phi_k)^2}$ , where  $\hat{\theta}_{k,n}$ ,  $\hat{\phi}_{k,n}$  are the estimated value of  $\theta_k$  and  $\phi_k$  for the  $n$ th trial.

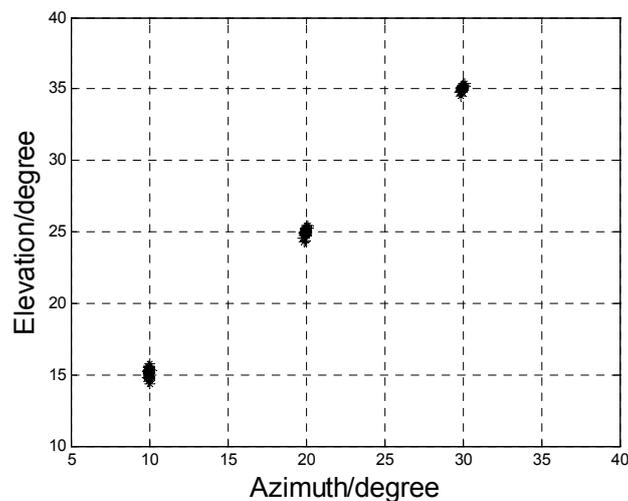
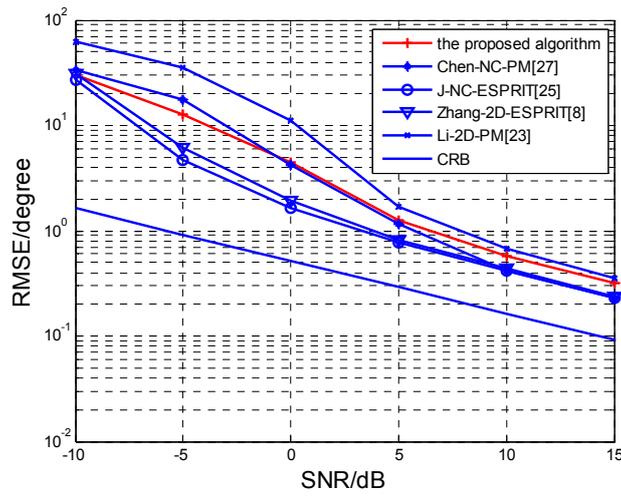
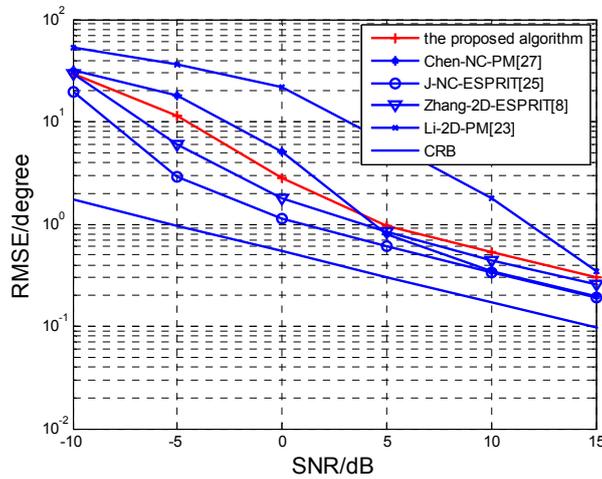


Figure 6. Angle estimation results.

Figure 7a,b presents RMSE comparison at different SNRs among the proposed algorithm, J's NC-ESPRIT algorithm [25], Chen's NC-PM algorithm [27], Zhang's 2D-ESPRIT algorithm [8], Li's 2D-PM algorithm [23] and CRB. In Figure 5a, we set  $M = 6$ ,  $N = 8$ ,  $L = 100$ . In Figure 5b, we change the numbers of sensors and snapshots and set  $M = 8$ ,  $N = 8$ , and  $L = 50$ . From the curves of Figure 5a,b, we know that the proposed algorithm has better RMSE performance than Li's algorithm [23]. Furthermore, it has close RMSE performance to Chen's algorithm [27]. However, we should know that our algorithm has much lower computational amount than J's NC-ESPRIT algorithm and Chen's NC-PM algorithm owing to the real-valued processing, which means that it is more suitable for a practical application system.

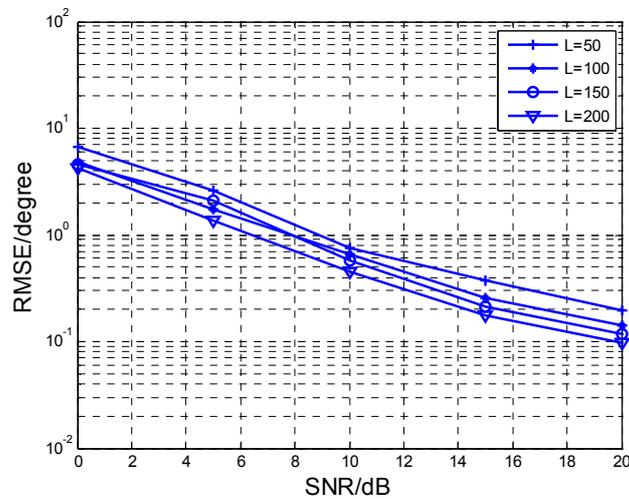


(a)



(b)

**Figure 7.** The root mean squared error (RMSE) comparison of different algorithms versus SNR. (a)  $M = 6, N = 8, L = 100$ ; (b)  $M = 8, N = 8, L = 50$ .



**Figure 8.** RMSE comparison at different values of  $L$ .

Figure 8 presents RMSE performance comparisons at different snapshots  $L$ . Where  $M = 8$ ,  $N = 6$ , SNR is varied from 0 dB to 20 dB. We can observe that the RMSE performance is improved with the increase of snapshot  $L$ . When  $L$  increases, we get more samples to estimate the propagator matrix more accurately, and so the angle estimation performance is enhanced.

Figures 9 and 10 present RMSE versus different values of  $M$  or  $N$ , respectively. The snapshot  $L$  is fixed at 200. In addition, it is indicated that RMSE performance is improved when  $M$  or  $N$  increases. Multiple sensors enhance the aperture of the array as well as diversity gain. Therefore, it can improve the angle estimation performance.

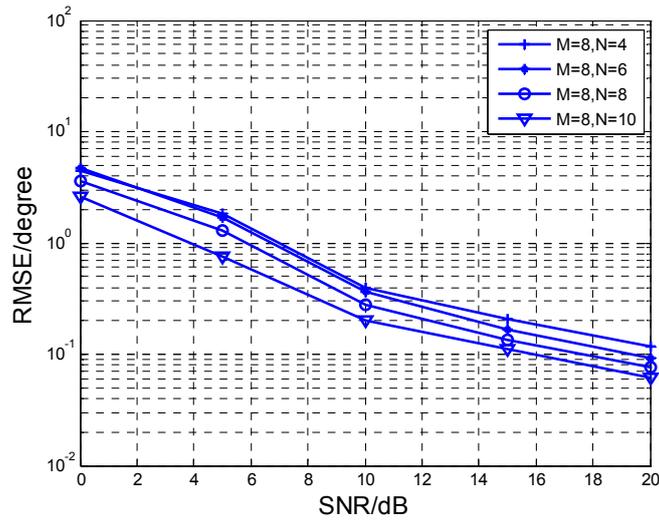


Figure 9. RMSE comparison at different  $N$  with  $M = 8$ .

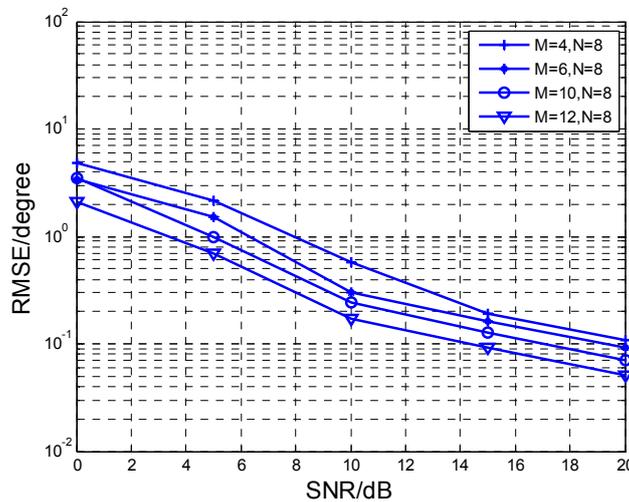


Figure 10. RMSE comparison at different  $M$  with  $N = 8$ .

The estimation performance for two closely spaced sources is also investigated. Figure 11 depicts the scatter plot of 2D-DOA estimation results for two closely spaced sources. Where  $M = 8$ ,  $N = 10$ , SNR = 10 dB, the snapshot  $L$  is 200. It is shown that our algorithm works well for the closely spaced sources.

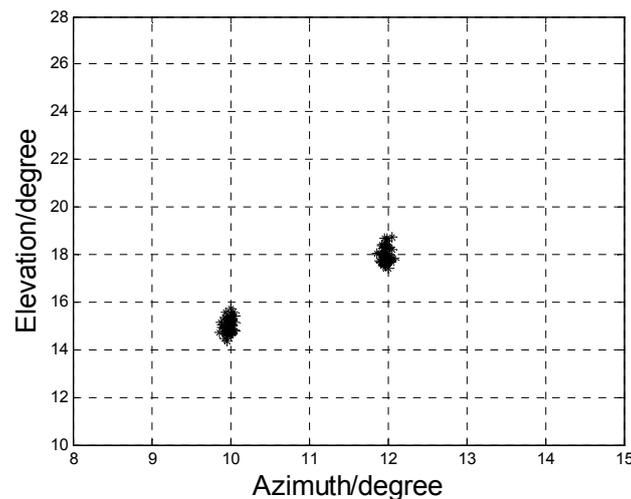


Figure 11. Scatter plot with closely spaced sources.

## 6. Conclusions

We have presented a novel direction finding algorithm for uniform rectangular planar array. The characteristics of noncircular signal and Euler's transformation are exploited to get the real-valued rectangular array data in a new way. The proposed algorithm can reduce the computational amount since it does not refer to plural operation and the eigenvalues' decomposition of the covariance matrix. The theory analysis and simulation results verify that our algorithm is more suitable for real-time processing system in engineering.

**Acknowledgments:** This work is supported by China Scholarship Council Grants (201606835025), National Natural Science Foundation of China (61371169) and the Electronic & Information School of Yangtze University Innovation Foundation (2016-DXCX-05). Xiaofei Zhang participated the design of the algorithm. He gave the paper many suggestions.

**Author Contributions:** Lingyun Xu wrote the paper and revised the manuscript. Fangqing Wen carried out the simulation. All authors read and approved the final manuscript.

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

## Abbreviations

2D-DOA	Two-dimensional direction of arrival
URA	Uniform rectangular planar array
PM	Propagator method
CRB	Crame–Rao bound
RMS	Real-valued multiplications
RMSE	Root mean square error
SNR	Signal-to-noise ratio

## References

1. Min, S.; Seo, D.; Lee, K.B.; Kwon, H.M.; Lee, Y.H. Direction-of-arrival tracking scheme for DS/CDMA systems: Direction lock loop. *IEEE Trans. Wirel. Commun.* **2004**, *3*, 191–202. [[CrossRef](#)]
2. Chiang, C.T.; Chang, A.C. DOA estimation in the asynchronous DS-CDMA system. *IEEE Antennas Propag.* **2003**, *51*, 40–47. [[CrossRef](#)]
3. Yang, S.; Li, Y.; Zhang, K.; Tang, W. Multiple-parameter estimation method based on spatio-temporal 2-D processing for bistatic MIMO radar. *Sensors* **2015**, *15*, 31442–31452. [[CrossRef](#)] [[PubMed](#)]
4. Zhang, X.F.; Zhou, M.; Li, J.F. A PARALIND decomposition-based coherent two-dimensional direction of arrival estimation algorithm for acoustic vector-sensor arrays. *Sensors* **2013**, *13*, 5302–5316. [[CrossRef](#)] [[PubMed](#)]

5. Liang, H.; Cui, C.; Dai, L.; Yu, J. Reduced-dimensional DOA estimation based on ESPRIT algorithm in MIMO radar with L-shaped array. *J. Electron. Inf. Technol.* **2015**, *37*, 1828–1835.
6. Wax, M.; Shan, T.J.; Kailath, T. Spatio-temporal spectral analysis by eigenstructure methods. *IEEE Trans. Acoust. Speech Signal Process.* **1984**, *32*, 817–827. [[CrossRef](#)]
7. Zoltowski, M.D.; Haardt, M.; Mathews, C.P. Closed-form 2-D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT. *IEEE Trans. Signal Process.* **1996**, *44*, 316–328. [[CrossRef](#)]
8. Zhang, X.; Gao, X.; Chen, W. Improved blind 2D-direction of arrival estimation with L-shaped array using shift invariance property. *J. Electromagn. Waves Appl.* **2009**, *23*, 593–606. [[CrossRef](#)]
9. Del Rio, J.E.F.; Catedra-Perez, M.F. The matrix pencil method for two-dimensional direction of arrival estimation employing an L-shaped array. *IEEE Trans. Antennas Propag.* **1997**, *45*, 1693–1694. [[CrossRef](#)]
10. Clark, M.P.; Scharf, L.L. Two-dimensional modal analysis based on maximum likelihood. *IEEE Trans. Signal Process.* **1994**, *42*, 1443–1452. [[CrossRef](#)]
11. Fang, W.H.; Lee, Y.C.; Chen, Y.T. Maximum likelihood 2-D DOA estimation via signal separation and importance sampling. *IEEE Antennas Wirel. Propag. Lett.* **2016**, *16*, 746–749. [[CrossRef](#)]
12. Zhang, X.F.; Li, J.F.; Xu, L.Y. Novel two-dimensional DOA estimation with L-shaped array. *EURASIP J. Adv. Signal Process.* **2011**, *50*, 1–7. [[CrossRef](#)]
13. Nie, X.; Li, L.P. A computationally efficient subspace algorithm for 2-D DOA estimation with L-shaped array. *IEEE Signal Process. Lett.* **2014**, *21*, 971–974.
14. Wang, G.M.; Xin, J.M.; Zheng, N.N.; Sano, A. Computationally efficient subspace-based method for two-dimensional direction estimation with L-shaped array. *IEEE Trans. Signal Process.* **2011**, *59*, 3197–3212. [[CrossRef](#)]
15. Wei, Y.S.; Guo, X.J. Pair-matching method by signal covariance matrices for 2D-DOA estimation. *IEEE Antennas Wirel. Propag. Lett.* **2014**, *13*, 1199–1202.
16. Zhang, W.; Liu, W.; Wang, J.; Wu, S.L. Computationally efficient 2D DOA estimation for uniform rectangular arrays. *Multidimens. Syst. Signal Process.* **2014**, *25*, 847–857. [[CrossRef](#)]
17. Yu, H.X.; Qiu, X.F.; Zhang, X.F. Two-dimensional direction of arrival estimation for rectangular array via compressive sensing trilinear model. *Int. J. Antenna Propag.* **2015**, *2015*, 297572. [[CrossRef](#)]
18. Xu, X.; Ye, Z. Two-dimensional direction of arrival estimation by exploiting the symmetric configuration of uniform rectangular array. *IET Radar Sonarav. Navig.* **2012**, *6*, 307–313. [[CrossRef](#)]
19. Wu, H.; Hou, C.P.; Chen, H.; Liu, W.; Wang, Q. Direction finding and mutual coupling estimation for uniform rectangular arrays. *Signal Process.* **2015**, *117*, 61–68. [[CrossRef](#)]
20. Gu, J.F.; Zhu, W.P.; Swamy, M.N.S. Joint 2D DOA estimation via sparse L-shaped array. *IEEE Trans. Signal Process.* **2015**, *63*, 1171–1182. [[CrossRef](#)]
21. Marcos, S.; Marsal, A.; Benider, M. The propagator method for sources bearing estimation. *Signal Process.* **1995**, *42*, 121–138. [[CrossRef](#)]
22. Wu, Y.; Liao, G.S.; So, H.C. A fast algorithm for 2-D direction of-arrival estimation. *Signal Process.* **2003**, *83*, 1827–1831. [[CrossRef](#)]
23. Li, J.F.; Zhang, X.F.; Chen, H. Improved two-dimensional DOA estimation algorithm for two-parallel uniform linear arrays using propagator method. *Signal Process.* **2012**, *92*, 3032–3038. [[CrossRef](#)]
24. Abeida, H.; Delmas, J.P. MUSIC-like estimation of direction of arrival for noncircular sources. *IEEE Trans. Signal Process.* **2006**, *54*, 2678–2690. [[CrossRef](#)]
25. Steinwandt, J.; Roemer, F.; Haardt, M. Performance analysis of ESPRIT-type algorithms for non-circular sources. In Proceedings of the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 13), Vancouver, BC, Canada, 26–31 May 2013; pp. 3986–3990.
26. Zhang, L.; Lv, W.H.; Zhang, X.F. 2D-DOA estimation of noncircular signals for uniform rectangular array via NC-PARAFAC method. *Int. J. Electron.* **2016**, *103*, 1839–1856. [[CrossRef](#)]
27. Chen, C.Q.; Wang, C.H.; Zhang, X.F. DOA and Noncircular phase estimation of noncircular signal via an improved noncircular rotational invariance propagator method. *Math. Probl. Eng.* **2015**, *2015*, 235173. [[CrossRef](#)]

28. Picinbono, B. On circularity. *IEEE Trans. Signal Process.* **1994**, *42*, 3473–3482. [[CrossRef](#)]
29. Stoica, P.; Nehorai, A. Performance study of conditional and unconditional direction-of-arrival estimation. *IEEE Trans. Acoust. Speech Signal Process.* **1990**, *38*, 1783–1795. [[CrossRef](#)]



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).