

# Virtual Linear Array Modelling of a Planar Array

Marc Willerton and Athanassios Manikas

Communication and Signal Processing Group,  
Department of Electrical and Electronic Engineering,  
Imperial College London, UK.

{marc.willerton05, a.manikas}@imperial.ac.uk

**Abstract**—In this paper, a novel transformation connecting an arbitrary Planar Array to a virtual Uniform Linear Array (ULA) with a larger number of sensors is proposed. The array orientation, number of sensors and phase characteristics of the virtual array are a function of the array shape of the planar array. The two parameter manifold (azimuth, elevation) of a planar array of  $N$  sensors is a conoid surface embedded into an  $N$ -dimensional complex space which can be described by two families of curves ( $\theta$ -curves and  $\phi$ -curves). While the family of  $\phi$ -curves have hyperhelical shape, the family of  $\theta$ -curves do not. The proposed transformation will allow the  $\theta$ -curves to be transformed to the manifold of a virtual ULA that has hyperhelical shape. This will allow a planar array system to be analysed or designed by analysing or designing simple hyperhelical curves associated with the virtual linear array.

## Notation

$A, a$	Scalar
$\underline{A}, \underline{a}$	Column vector
$\mathbb{A}$	Matrix
$(\cdot)^T$	Transpose
$(\cdot)^H$	Hermitian Transpose
$\odot$	Hadamard product
$\oslash$	Hadamard division
$\exp(\underline{A})$	Element-by-element exponential of the vector $\underline{A}$
$J_n(a)$	Bessel of the first kind of order $n$ and argument $a$
$J_n(\underline{a})$	Element-by-element Bessel function of the vector $\underline{a}$
$\underline{a}^b$	Element by element power of the vector $\underline{a}$
$\underline{0}_N$	Column vector of $N$ zeros
$\underline{1}_N$	Column vector of $N$ ones
$\mathcal{R}$	Set of real numbers
$\mathcal{C}$	Set of complex numbers

## I. INTRODUCTION

An array system is defined as a collection of sensors which are distributed in 3 dimensional real space with a common reference point. Array signal processing is concerned with the exploitation of this spatial diversity to solve three main problems. These are namely signal detection, source parameter estimation and signal reception (i.e. beamforming)

which are all employed in a wide variety of military and civilian applications in both RF and acoustic environments. Examples include Towed Arrays, Multiple Input Multiple Output (MIMO) Radar, Synthetic Aperture Radar and Arrayed Wireless Sensor Networks. The distribution of the array sensors (i.e. the array geometry) directly influences the overall capabilities of any array processing system. Hence, it is crucial to fully understand the geometrical aspects of such a system. Consider an array of  $N$  omnidirectional sensors with sensor locations given in Cartesian coordinates by the  $N \times 3$  matrix in units of half-wavelengths

$$\begin{bmatrix} r_x & r_y & r_z \end{bmatrix} \in \mathcal{R}^{N \times 3} \quad (1)$$

The array response vector, or array manifold vector of a plane wave arriving from azimuth angle  $\theta$  and elevation  $\phi$  is modelled as

$$\underline{S}(\theta, \phi) = \exp(-j \begin{bmatrix} r_x & r_y & r_z \end{bmatrix} \underline{k}(\theta, \phi)) \quad (2)$$

where  $\underline{k}(\theta, \phi)$  is the wavenumber vector which is defined in metres ( $m$ ) and units of half-wavelengths ( $\frac{\lambda}{2}$ ) as

$$\underline{k}(\theta, \phi) = \frac{2\pi}{\lambda} [\cos \theta \cos \phi, \sin \theta \cos \phi, \sin \phi]^T \text{ in } m \quad (3a)$$

$$= \pi [\cos \theta \cos \phi, \sin \theta \cos \phi, \sin \phi]^T \text{ in } \frac{\lambda}{2} \quad (3b)$$

In Equation 3a,  $\lambda$  is the wavelength of the planewave (e.g. carrier) and  $c$  denotes its propagation speed. The manifold vector in Equation 2 contains all the information about the array geometry when a planewave of wavelength  $\lambda$  is incident on the array from the direction  $(\theta, \phi)$ . Note that since the source is operating in the far field of the array, the manifold vector has no dependence on the range  $\rho$  of the source from the array reference point.

In Equation 2,  $\underline{S}(\theta, \phi)$  denotes the manifold vector of an array of  $N$  sensors where  $(\theta, \phi)$  are directional parameters. As these parameters vary, the point  $\underline{S}$  will trace out a surface embedded in an  $N$  dimensional complex space  $\mathcal{C}^N$ . This is known as the array manifold which is a surface  $\mathcal{M}$  embedded into  $N$ -dimensional complex space and is formally defined as

$$\mathcal{M} \triangleq \{ \underline{S}(\theta, \phi) \in \mathcal{C}^N, \forall (\theta, \phi) : \theta, \phi \in \Omega \} \quad (4)$$

where  $\Omega$  denotes the parameter space of the source direction which in the most general case is

$$\Omega = \{ (\theta, \phi) : \theta \in [0^\circ, 360^\circ) \text{ and } \phi \in (-90^\circ, 90^\circ) \} \quad (5)$$

This research was funded by the University Defence Research Centre (UDRC) in Signal Processing (MOD, UK)

Schmidt states in [1] that the array manifold fully describes the array system and its capabilities. According to [2], this makes it possible to analyse or design an array system by analysing or designing the "mathematical object" associated with the array manifold. In turn this illustrates the importance of understanding the array manifold within array processing. For example, there may be a requirement for the array to have a good resolution performance in a certain source direction whilst still being able to operate adequately in others. In this case, an understanding of the effect of array geometry on the properties of the manifold will allow an appropriate array to be designed (for example see [3]). Furthermore, given an array geometry, an understanding of the corresponding manifold allows array ambiguities to be uncovered. This indicates where the mapping from the parameter space to the manifold is not one to one which is crucial to understand when analysing data received from an array system (see Chapter 6 in [2] for more information).

Differential geometry holds the key to characterising the properties of the array manifold. However, this only provides local properties of a complex shape which in general makes it difficult to study in its entirety. Hyperhelices are special curves in differential geometry which can be fully described by a set of curvatures allowing local properties to be applied globally. For example, a hyperhelix embedded into  $N$ -dimensional complex space can be fully described by at most  $2N$  curvatures which remain fixed at any point on the curve. The manifold described in Equation 4 is a surface which may be fully described by two families of curves. These are namely the  $\theta$ -curves and the  $\phi$ -curves. For a planar array geometry, the  $\phi$ -curves will all be hyperhelices but the  $\theta$ -curves will not be. This makes the properties of the  $\theta$ -curves difficult to characterise. In contrast, a linear array geometry creates an array manifold which is a curve instead of a surface and always has hyperhelical shape. In this paper, a transformation connecting a planar array and a set of virtual ULA's (one for each  $\theta$ -curve) with a much larger number of sensors will be presented. This will allow the  $\theta$ -curves for a planar array which are not hyperhelices to be mapped to hyperhelices in a much larger complex space.

Initially, in Section II, the array signal model for the planar and linear array geometries will be developed from the general model in Equation 2. The properties of their manifolds will also be discussed in terms of differential geometry. In Section III, the transformation between the Planar Array and the virtual ULA will be developed. In Section ??, an analysis of the properties of the virtual array produced by the proposed transformation for a number of planar array geometries will be given as well as a representative example proving the correctness and applicability of the transformation. Finally, in Section V, the paper will be concluded.

## II. PLANAR AND LINEAR ARRAY SIGNAL MODELS

Consider a linear array geometry of  $N$  omnidirectional sensors ( $r_y = \underline{0}_N$  and  $r_z = \underline{0}_N$ ). The manifold vector in contrast

to the generalised form in Equation 2 becomes

$$\underline{S}(\theta) = \exp(-j\pi r_x \cos \theta) \quad (6)$$

Note that now Equation 6 spans only one parameter  $\theta$  and creates a curve  $\mathcal{A}$  embedded into  $N$ -dimensional complex space  $\mathcal{C}^N$  formally defined as

$$\mathcal{A} \triangleq \{\underline{S}(\theta) \in \mathcal{C}^N, \forall \theta : \theta \in \Omega\} \quad (7)$$

where  $\Omega$  denotes the parameter space which in contrast to Equation 5 is reduced to

$$\Omega = \{\theta : \theta \in [0^\circ, 180^\circ)\} \quad (8)$$

In [2], it is shown that the manifold of a linear array of sensors with a uniform or non-uniform, symmetric or asymmetric geometry is a hyperhelical curve embedded into  $N$ -dimensional complex space. The number of dimensions spanned by the hyperhelix is determined by the number of symmetrical sensors about the array centroid [4]. In general, a space curve can be fully represented by a set of numbers (known as curvatures) for each point on the curve that fully describes it. Consequently, these curvatures describe the array of sensors and its performance. However, hyperhelices are very important and useful space curves since these curvatures remain constant at different points on the curve and hence are easier to characterise. Symmetric linear arrays are a subset of linear arrays which produce hyperhelices which are said to stand upright (zero inclination angle). This implies that the manifold exists entirely in  $\mathcal{R}^N$  dimensions of the complex space.

In a planar array geometry of  $N$  omnidirectional sensors ( $r_z = \underline{0}_N$ ), the manifold vector in comparison to Equation 6 becomes more complex

$$\underline{S}(\theta, \phi) = \exp(-j\pi \cos \phi (r_x \cos \theta + r_y \sin \theta)) \quad (9)$$

Note that now Equation 9 spans two parameters ( $\theta, \phi$ ) in its parameter space and hence creates a surface  $\mathcal{M}$  embedded in  $N$  dimensional complex space  $\mathcal{C}^N$ . This creates a two parameter manifold consistent with the formal definition provided in Equation 4. The parameter space  $\Omega$  for a planar array is larger than for the linear array as described in Equation 5 but more restrictive than the general case as described in Equation 8.

$$\Omega = \{(\theta, \phi) : \theta \in [0^\circ, 360^\circ) \text{ and } \phi \in [0^\circ, 90^\circ)\} \quad (10)$$

In [5], it has been proven that the manifold surface of a planar array of  $N$  omnidirectional sensors is a conoid embedded into  $N$  dimensional complex space. Furthermore, it has been proven that this surface can be mapped to a real plane without any loss of information. The surface can be described by two families of curves. These are namely the  $\theta$ -curves ( $\phi = \text{constant}$ ) and the  $\phi$ -curves ( $\theta = \text{constant}$ ). In [6], it has been proven that  $\phi$ -curves are complex hyperhelices whilst  $\theta$ -curves are not. This implies that the local properties of the  $\theta$ -curves change, making them more complex to characterise. An alternative is using the "cone" angles which

provides  $\alpha$ -curves and  $\beta$ -curves which are both hyperhelical (see Chapter 5 in [2] for more information) and hence can describe the manifold surface by a set of constant curvatures which is valid at any point on the manifold. This allows the analysis performed for the manifold of linear array geometries to be applied to these curves to characterise the planar array manifold.

In this paper, a novel transformation that connects a  $\theta$ -curve of a Planar Array to a hyperhelical curve representing the manifold of a virtual ULA with a larger number of sensors is proposed. This allows the  $\theta$ -curves of a planar array embedded into  $N$  dimensional complex space to now be expressed as hyperhelical curves lying with zero inclination angle in a larger complex space. In turn, this will allow planar array systems to be fully analysed or designed using the framework developed for linear arrays. The virtual ULA has a number of sensors, orientation and phase characteristics which describes the shape of the planar array. The proposed transformation connecting the virtual and planar manifold vectors is a function of the planar array geometry and the  $\phi$  parameter but is independent of  $\theta$ . In the next section of this paper, this transformation will be developed.

### III. VIRTUAL LINEAR ARRAY TRANSFORMATION

Consider a planar array of  $N$  omnidirectional sensors described by the  $N \times 3$  matrix in units of half-wavelengths

$$\begin{bmatrix} r_x & r_y & \underline{0}_N \end{bmatrix} \quad (11)$$

The corresponding manifold vector describing the response of a plane wave arriving from (*azimuth, elevation*) =  $(\theta, \phi)$  is given by

$$\underline{S}(\theta, \phi) = \exp(-j\pi \cos \phi (r_x \cos \theta + r_y \sin \theta)) \quad (12)$$

which may be re-written as follows

$$\underline{S}(\theta, \phi) = \exp(-j\pi \underline{R}(\phi) \odot \cos(\theta \underline{1}_N - \underline{\psi})) \quad (13)$$

where

$$\begin{aligned} \underline{R}(\phi) &= [R_1(\phi), R_2(\phi), \dots, R_N(\phi)]^T \\ &= \sqrt{r_x^2 + r_y^2} \cos(\phi) \end{aligned} \quad (14)$$

$$\begin{aligned} \underline{\psi} &= [\psi_1, \psi_2, \dots, \psi_N]^T \\ &= \tan^{-1}(r_y \oslash r_x) \end{aligned} \quad (15)$$

With reference to [7], page 795, a complex exponential can be expanded as an infinite summation of Bessel functions  $J_n(z)$

$$\exp(jz \odot \sin(A)) = \sum_{q=-\infty}^{\infty} \{J_q(z) \odot \exp(jqA)\} \quad (16)$$

Thus using Equation 16 with

$$z = -\pi \underline{R}(\phi) \quad (17)$$

$$\underline{A} = \left(\theta + \frac{\pi}{2}\right) \underline{1}_N - \underline{\psi} \quad (18)$$

Equation 13 can be re-written as follows

$$\begin{aligned} \underline{S}(\theta) &= [S_1(\theta), S_2(\theta), \dots, S_N(\theta)]^T \\ &= \sum_{q=-\infty}^{\infty} J_q(-\pi \underline{R}(\phi)) \\ &\quad \odot \exp\left(jq \left(\left(\theta + \frac{\pi}{2}\right) \underline{1}_N - \underline{\psi}\right)\right) \end{aligned} \quad (19)$$

where  $J_q(-\pi \underline{R}(\phi))$  represents a real vector of Bessel functions with its  $i^{th}$  element  $J_q(-\pi R_i(\phi))$  a scalar Bessel function of first kind with integer order  $q$ . That is

$$J_q(-\pi \underline{R}(\phi)) = \begin{bmatrix} J_q(-\pi R_1(\phi)) \\ J_q(-\pi R_2(\phi)) \\ \vdots \\ J_q(-\pi R_N(\phi)) \end{bmatrix} \in \mathcal{R}^N \quad (20)$$

However,

$$\text{As } |q| > |\pi R_i(\phi)|, J_q(-\pi R_i(\phi)) \rightarrow 0, \forall i \quad (21)$$

Hence, a good approximation to Equation 19 can be formed by only summing between  $-Q$  and  $Q$  instead of  $-\infty$  and  $\infty$ . This approximation will introduce an error vector  $\underline{\varepsilon}$  which is defined as

$$\underline{\varepsilon} = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_N]^T$$

Using this, Equation 19 becomes

$$\begin{aligned} \underline{S}(\theta) &= \sum_{q=-Q}^Q J_q(-\pi \underline{R}(\phi)) \\ &\quad \odot \exp\left(jq \left(\left(\theta + \frac{\pi}{2}\right) \underline{1}_N - \underline{\psi}\right)\right) + \underline{\varepsilon} \end{aligned} \quad (22)$$

Let us define the matrix  $\mathbb{B} \in \mathcal{R}^{N \times (2Q+1)}$  as

$$\begin{aligned} \mathbb{B} &\triangleq [\underline{B}_1, \underline{B}_2, \dots, \underline{B}_N]^T \\ &= [J_{-Q}(-\pi \underline{R}(\phi)), J_{-Q+1}(-\pi \underline{R}(\phi)), \dots \\ &\quad \dots, J_{+Q-1}(-\pi \underline{R}(\phi)), J_{+Q}(-\pi \underline{R}(\phi))] \end{aligned} \quad (23)$$

where

$$\underline{B}_i = \begin{bmatrix} J_{-Q}(-\pi R_i(\phi)) \\ J_{-Q+1}(-\pi R_i(\phi)) \\ \vdots \\ J_{+Q-1}(-\pi R_i(\phi)) \\ J_{+Q}(-\pi R_i(\phi)) \end{bmatrix} \in \mathcal{R}^{(2Q+1) \times 1} \quad (24)$$

Then

$$\underline{S}(\theta) = \begin{bmatrix} \underline{B}_1^T \exp(jr_Q(\theta + \frac{\pi}{2} - \psi_1)) \\ \underline{B}_2^T \exp(jr_Q(\theta + \frac{\pi}{2} - \psi_2)) \\ \vdots \\ \underline{B}_N^T \exp(jr_Q(\theta + \frac{\pi}{2} - \psi_N)) \end{bmatrix} \quad (25)$$

where

$$r_Q = [-Q, -Q+1, \dots, +Q-1, Q]^T \in \mathcal{R}^{2Q+1} \quad (26)$$

denotes the locations of the virtual array sensors in units of half-wavelength. i.e.

$$S_i(\theta) = \underline{B}_i^T \exp(jr_Q(\theta + \frac{\pi}{2} - \psi_i)) + \varepsilon_i \quad (27)$$

The following azimuth transformation is defined to allow the field of view of the planar array to be transformed to that of a linear array

$$\theta = \pi - \pi \cos(\theta_v) + \underline{1}_N^T \underline{\psi} \quad (28)$$

where  $\theta_v$  represents the azimuth angle of the source in the field of view of the virtual linear array. Note that the transformation implies that  $\theta_v = 0^\circ$  corresponds to  $\theta = \underline{1}_N^T \underline{\psi}$ . Equation 27, now becomes

$$S_i(\theta_v) = \left( \underline{B}_i^T \odot \exp(-j \underline{r}_Q^T \underline{\psi}_i) \right) \cdot \exp \left( -j \left( \pi \underline{r}_Q \cos \theta_v + \underline{r}_Q \left( \frac{\pi}{2} - \underline{1}_N^T \underline{\psi} \right) \right) \right) + \varepsilon_i \quad (29)$$

$$= \left( \underline{B}_i \odot \exp(-j \underline{r}_Q \underline{\psi}_i) \right)^T \cdot \exp(-j(\pi \underline{r}_Q \cos \theta_v + \underline{u})) + \varepsilon_i \quad (30)$$

Combining these expressions for each of the  $N$  sensors in the planar array, Equation 25 becomes

$$\underline{S}(\theta) = \left( \begin{bmatrix} \underline{B}_1^T \\ \underline{B}_2^T \\ \vdots \\ \underline{B}_N^T \end{bmatrix} \odot \exp(-j \underline{\psi} \underline{r}_Q^T) \right) \cdot \exp(-j(\pi \underline{r}_Q \cos \theta_v + \underline{u})) + \underline{\varepsilon} \quad (31)$$

$$= \underbrace{\left( \underline{B}^T \odot \exp(-j \underline{\psi} \underline{r}_Q^T) \right)}_{\triangleq \mathbb{T} \in \mathcal{C}^{N \times (2Q+1)}} \cdot \underbrace{\exp(-j(\pi \underline{r}_Q \cos \theta_v + \underline{u}))}_{\triangleq \underline{S}_v(\theta_v) \in \mathcal{C}^{(2Q+1) \times 1}} + \underline{\varepsilon} \quad (32)$$

Hence

$$\underline{S}(\theta) = \mathbb{T} \underline{S}_v(\theta_v) + \underline{\varepsilon} \quad (33a)$$

where

$$\mathbb{T}(\phi) = \left( \underline{B}^T \odot \exp(-j \underline{\psi} \underline{r}_Q^T) \right) \quad (33b)$$

$$\underline{S}_v(\theta_v) = \exp(-j(\pi \underline{r}_Q \cos \theta_v + \underline{u})) \quad (33c)$$

Here,  $\underline{S}_v(\theta_v)$  has the form of the general expression of a hyperhelix. This corresponds to a virtual array of  $N_v$  virtual sensors where  $N_v = 2Q+1$ . The virtual sensors are located on a line which is at an angle  $\underline{1}_N^T \underline{\psi}$  anticlockwise from the positive x-axis associated with the planar array at locations described by the  $N_v \times 1$  vector  $\underline{r}_Q$  in units of half-wavelengths. Since  $\underline{r}_Q$  is made of integers ranging from  $-Q$  to  $Q$ , it is clear that a virtual ULA with half unit wavelength inter-sensor spacing is formed. Furthermore, the array reference point of the virtual array will be at the reference point of the planar array which will also be the array centroid of the virtual array. Hence, the virtual array will be fully symmetric. Each virtual sensor has a corresponding phase which is described by the  $N_v \times 1$

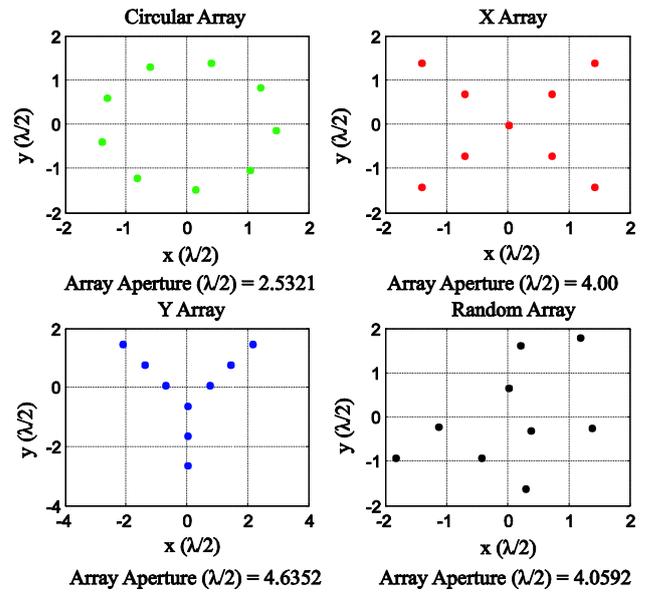


Fig. 1. The planar array geometries to be analysed

vector  $\underline{u}$ . The number of virtual sensors, their electrical phase characteristics and the orientation of the array (and hence the hyperhelix described by different curvatures) is a function of the geometry of the planar array. Note that the virtual manifold vector  $\underline{S}_v$  is independent of the  $\phi$  parameter. The transformation matrix  $\mathbb{T}$  connecting the planar and linear array manifold vectors is a function of the planar array geometry and the  $\phi$  parameter but is independent of  $\theta$ . Knowing the geometry of the planar array, the virtual linear array and its corresponding array manifold can be deduced. This approach could also be extended to provide a transformation connecting a 3D array to a virtual linear array.

#### IV. REPRESENTATIVE EXAMPLES

In this section, the transformation developed in Section III will be applied to four different omnidirectional planar array geometries of  $N = 9$  sensors and an analysis of the resulting virtual array properties will be performed. Following this, a representative example illustrating an application for the transformation will be given. Initially, consider the planar array geometries given in Figure 1. Note that following Equations 14 and 21,  $\underline{R}(\phi)$  and hence  $N_v$  is dependent on the location of the planar array reference point. However, the location of this point is arbitrary and doesn't affect the performance of the array system. Therefore its location should not affect the properties of the virtual array either. Hence, for each of the planar array geometries in Figure 1, the array reference point is fixed at the array centroid to overcome this issue. Furthermore, note that each of the planar arrays have  $\frac{\lambda}{2}$  intersensor spacing resulting in each geometry having a different array aperture as illustrated in Figure 1.

As stated previously, the planar array manifold can be expressed as two families of curves - the  $\theta$ -curves and  $\phi$ -curves. The transformation derived in the previous section

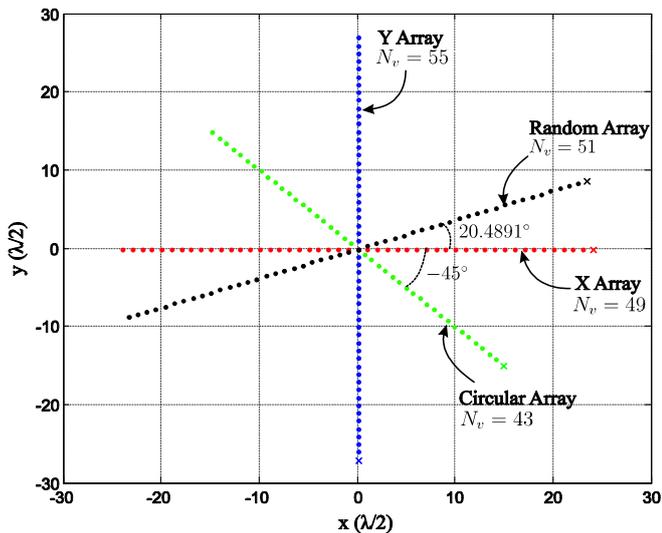


Fig. 2. The virtual Uniform Linear Arrays corresponding to the Planar Array geometries in Figure 1

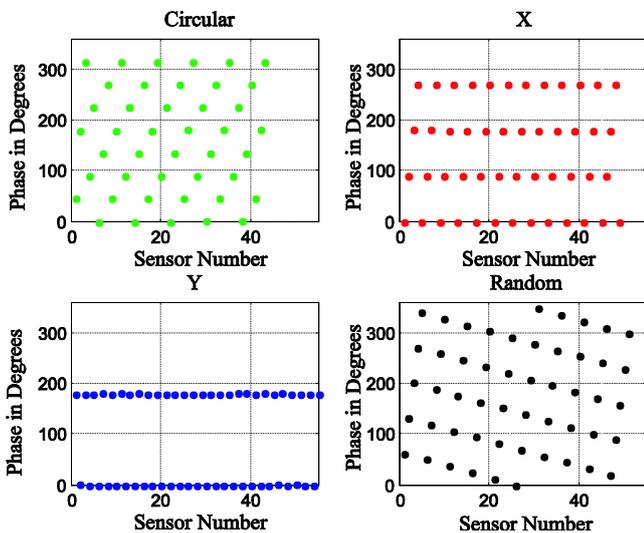


Fig. 3. Phase response of the virtual sensors in Figure 2

produces a virtual ULA for each  $\theta$ -curve (fixed  $\phi$ ). Initially take the  $\theta$ -curve when  $\phi = 0^\circ$ . Allowing a transformation error of  $\varepsilon < 10^{-12} \mathbf{1}_N$ , the virtual linear array geometries created using the proposed transformation are illustrated in Figure 2. Here, the sensors denoted with x represent the last sensor in the virtual array. The corresponding electrical phase characteristics for the virtual arrays are given in Figure 3.

Figure 2 shows that as the aperture of the planar array increases, the number of virtual sensors and hence the aperture of the virtual linear array (since virtual sensors have  $\frac{\lambda}{2}$  spacing) will increase also. This is intuitive since the array aperture is related to  $\underline{R}(\phi)$  and Equation 21 implies a larger number of virtual sensors will be required if  $\underline{R}(\phi)$  increases. Furthermore, the effect of the symmetry of the planar array upon the orientation of the virtual linear array as well as the electrical phase characteristics of the virtual array sensors

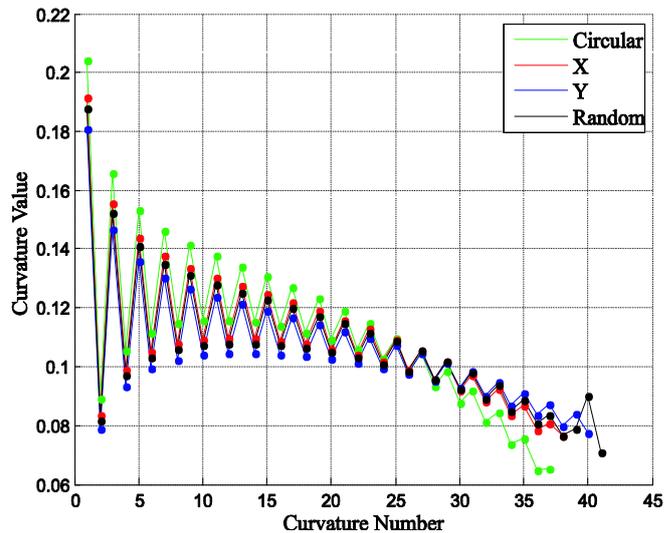


Fig. 4. Curvatures of the virtual Uniform Linear Arrays corresponding to the Planar Arrays in Figure 1

is also apparent. The X array which is fully symmetric in the  $x$  and  $y$  axes implies that  $\mathbf{1}_N^T \underline{\psi} = 0$ . This causes the corresponding virtual array to lie on the  $x$ -axis and the phase characteristics to be described by  $\underline{u} = \frac{\pi}{2} r_Q$  which will result in the virtual sensors taking one of only 4 unique values. The Y array which is symmetric in the  $y$  axis implies that  $\mathbf{1}_N^T \underline{\psi} = \pm \frac{\pi}{2}$ . This causes the corresponding virtual array to lie on the  $y$ -axis and the phase characteristics to be described by  $\underline{u} = \pm r_Q$  which will result in the virtual sensors taking one of only 2 unique values. The circular and random arrays which aren't symmetric in the  $x$  or  $y$  axis take other non-trivial values of  $\mathbf{1}_N^T \underline{\psi}$ . Hence, the corresponding virtual arrays lie at some orientation between 0 and  $2\pi$  and have phase characteristics which may take up to  $N_v$  unique values. Each of the virtual array geometries detailed above forms a hyperhelix in  $N_v$  dimensional complex space. Figure 4 gives the curvatures of these arrays.

The Multiple Signal Classification (MUSIC) algorithm [1] is a superresolution direction finding algorithm which partitions the observation space into signal and noise subspace - the space spanned by the columns of  $\mathbb{E}_s$  (the signal eigenvalues) or  $\mathbb{E}_n$  (the noise eigenvalues) respectively. Assuming a fixed elevation  $\phi$ , the MUSIC algorithm searches the corresponding  $\theta$ -curve. For a planar array the  $\theta$ -curve will not be hyperhelix. However, the signal or noise subspace can be transformed into virtual subspaces spanned by the columns of  $\mathbb{E}_{sv}$  or  $\mathbb{E}_{nv}$  where

$$\mathbb{E}_{sv} = \mathbb{T}^H \mathbb{E}_s \text{ and } \mathbb{E}_{nv} = \mathbb{T}^H \mathbb{E}_n \quad (34)$$

Note that in Equation 34,  $\mathbb{T}$  is the transformation matrix defined in Equation 33b. Then the hyperhelix of the virtual array manifold can be searched using the MUSIC cost function. For example, consider the circular array in Figure 1 in the presence of two sources at  $(\theta, \phi) = (150, 0)^\circ$  and  $(200, 0)^\circ$  assuming an infinite observation interval. Estimating the noise subspace and transforming this on to the virtual noise subspace using

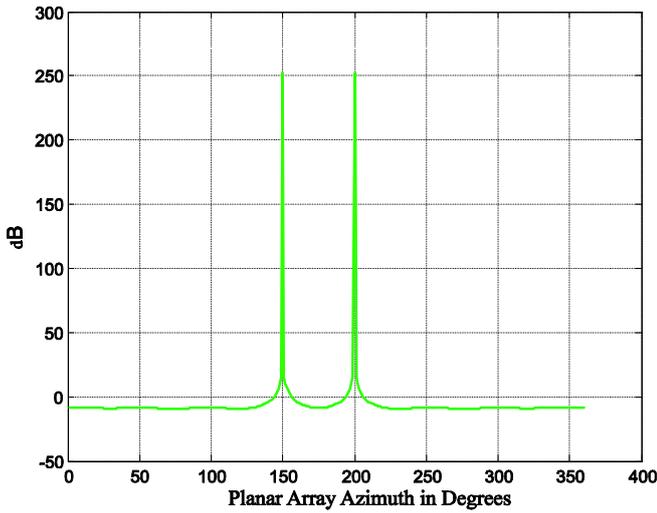


Fig. 5. MUSIC Spectrum produced by searching the Virtual Array corresponding to a UCA of  $N = 9$  sensors with sources at  $(\theta, \phi) = (150, 0)^\circ$  and  $(200, 0)^\circ$  under an infinite number of snapshots

Equation 34, the MUSIC spectrum produced by searching the virtual array manifold  $\underline{S}_v$ , corresponding to the virtual array geometry of the circular array in Figure 1, is shown in Figure 5. Within this example  $\mathbb{T}$  and  $\underline{S}_v$  are produced for  $\underline{\varepsilon} < 10^{-12} \underline{1}_N$ . Note also that since the manifold is now a hyperhelix, in this example root-MUSIC could have been applied to avoid an iterative search of the manifold.

In general, for each  $\theta$ -curve, a new virtual linear array and associated transformation matrix  $\mathbb{T}$  will be required to meet the fixed error bound  $\underline{\varepsilon}$ . However, since the only parameter to change between the different  $\theta$ -curves is  $\underline{R}(\phi)$ , only the number of sensors in each virtual array will change. Specifically, as the elevation angle increases, fewer virtual sensors  $N_v$  will be required to meet the error bound  $\underline{\varepsilon}$  and hence the virtual array aperture will reduce. The phase characteristics  $\underline{u}$  of the remaining sensors and the orientation of the array  $\underline{1}_N^T \underline{\psi}$  will remain the same in the virtual array corresponding to each of the  $\theta$ -curves. Furthermore, the structure of the transformation matrix  $\mathbb{T}$  will remain the same but will contain a different number of columns. It can be shown that the number of virtual sensors as a function of the azimuth angle (fixed  $\phi$ ), remains fixed for a fixed error bound  $\underline{\varepsilon}$ . However, the number of virtual sensors needed to represent each of the  $\theta$ -curves is not. This is illustrated in Figure 6 for each of the 4 planar array geometries constructed for the error bound  $\underline{\varepsilon} < 10^{-12} \underline{1}_N$ . Here, it is clear that for small elevation angles different numbers of virtual sensors are required for different array geometries. However, as  $\phi$  increases, smaller numbers of virtual sensors are required which tends to a fixed point independent of the planar array geometry. This is representative of the length of the  $\theta$ -curves in the planar array manifold which become smaller as  $\phi$  increases, due to the conoid shape of the manifold surface.

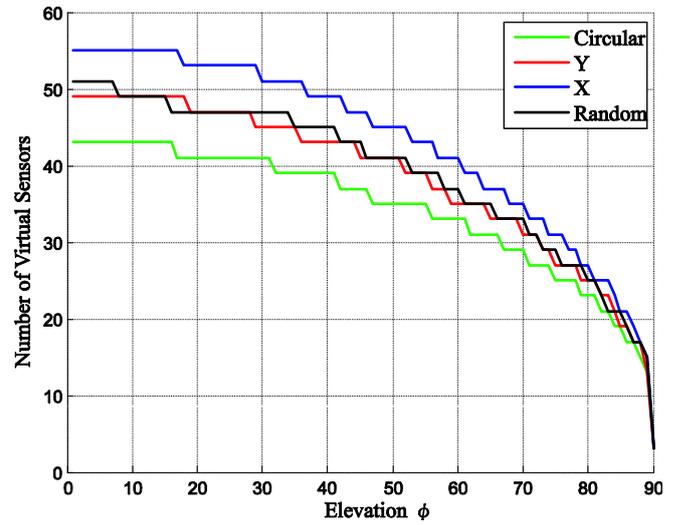


Fig. 6. Number of Virtual Sensors required to represent the  $\theta$ -curves for the 4 different planar array geometries to an error of  $\underline{\varepsilon} < 10^{-12} \underline{1}_N$ .

## V. CONCLUSIONS

In this paper a novel transformation given by Equation 33b connecting an arbitrary planar array to a virtual ULA has been presented based on the expansion of the planar array manifold vector using a summation of Bessel functions. This allows  $\theta$ -curves on the planar array manifold which in general are not hyperhelices to be analysed in a much larger complex space as a set of curves which are hyperhelices. Representative examples illustrate the properties of the virtual array produced for different planar array geometries. Furthermore, the correctness and applicability of the transform is shown by solving the MUSIC algorithm using a hyperhelical array manifold. It is shown how different  $\theta$ -curves produce virtual arrays with a differing number of virtual sensors relating to the length of the  $\theta$ -curve. This transformation has the potential to be used for the design and analysis of planar arrays.

## REFERENCES

- [1] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, Mar. 1986.
- [2] A. Manikas, *Differential Geometry in Array Processing*. Imperial College Press, 2004.
- [3] A. Manikas, A. Alexiou, and H. Karimi, "Comparison of the ultimate direction-finding capabilities of a number of planar array geometries," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 144, no. 6, p. 321, 1997.
- [4] N. Dowlut and A. Manikas, "Array Design for Superresolution Direction-Finding Algorithms," in *International Symposium on Digital Signal Processing*, 1996, pp. 38–43.
- [5] A. Manikas, A. Sleiman, and I. Dacos, "Manifold studies of nonlinear antenna array geometries," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 497–506, Mar. 2001.
- [6] H. Karimi and A. Manikas, "Manifold of a planar array and its effects on the accuracy of direction-finding systems," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 143, no. 6, p. 349, 1996.
- [7] J. Tuma and R. Walsh, *Engineering Mathematics Handbook*, 4th ed. McGraw-Hill, 1998.