VIRTUAL ARRAY DESIGN FOR ARRAY INTERPOLATION USING DIFFERENTIAL GEOMETRY

Markus Bühren *Marius Pesavento* *Johann F. Böhme*

Chair of System Theory and Signal Processing
Universität Stuttgart, D-70569 Stuttgart, Germany
markus.buehren@lss.uni-stuttgart.de

Dept. of Electrical Engineering and Information Sciences
Ruhr-Universität Bochum, D-44780 Bochum, Germany
{*mps, boehme*}*@sth.ruhr-uni-bochum.de*

ABSTRACT

In array interpolation, the optimal design of the virtual array geometry is still an open question. It is usually done heuristically by placing a virtual ULA into the center of the original array and fitting orientation and aperture by rule of thumb. In this paper, we parameterize the array manifold by its arc length and use this representation for the design of a virtual array manifold that optimally matches the directional properties of the original array. We verify the advantages of our new design method by simulation results and give some deeper understanding about the interrelation between the interpolation error, the condition number of the interpolation matrix and the DOA estimation bias.

1. INTRODUCTION

Arrays with special arrangements of sensors allow the application of simplified and computationally less demanding implementations of high-resolution direction finding methods. For example, the structure of single or multiple uniform linear arrays (ULA) is exploited in the formulation of the computationally efficient search-free root-MUSIC [1] and RARE [2] algorithms. Similarly, sensor arrays with shift-invariances facilitate search-free formulations of subspace methods like conventional and multiple invariance ESPRIT [3].

The idea of array interpolation techniques is to make search-free DOA estimation methods applicable to the general class of "non-structured" arrays. The original idea proposed by Friedlander [4] is a linear transformation of the real array manifold to a desired virtual ULA manifold over a preliminary defined directional sector. This method is bounded to small interpolation sectors and its DOA estimation performance is severely limited by a strong bias. Polynomial rooting needs to be performed for each interpolation sector individually, so the ability to choose large interpolation sectors is of high interest for the further reduction of computational effort.

The specification of the virtual ULA parameters, i.e. center position, orientation, number of sensors and inter-element spacing, is usually done heuristically. The aperture of the virtual array is chosen to be approximately as large as the original aperture, the virtual array is positioned into the center of the original array and the orientation is chosen perpendicular to the current interpolation sector or (if the original array has some specific orientation) like the original array.

In many cases, the directional properties of the original array and the virtual ULA differ significantly. For example, consider a plane circular-like arrangement of sensors. While the aperture size "seen" from different directions is nearly constant for the physical array, the effective aperture of the virtual linear array changes from the whole length of the array (seen from broadside) to zero (seen from the endfire directions). As the real and virtual arrays show a quite different directional behavior in this case, array interpolation over a large interpolation sector will obviously be a severe problem.

After introducing the signal model and briefly reviewing the principle of array interpolation, we will discuss in more detail some problems arising in the design of a virtual array. More specific, we will give some insight into the effect of the interpolation error and the condition number of the interpolation matrix on the DOA estimation bias. We will then introduce the arc length representation of the array manifold and use it for the development of a new procedure for the design of the virtual array manifold. A virtual Vandermonde structure is created that can be exploited by the root-MUSIC algorithm and optimally matches the directional properties of the original array. As in [5], the resulting virtual array can not be interpreted as a physically existing sensor array, but it has the mathematical properties needed for the application of several efficient DOA estimation algorithms.

2. ARRAY SIGNAL MODEL

In this paper, we use the common signal model

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\,\mathbf{s}(t) + \mathbf{n}(t), \tag{1}$$

where $\mathbf{s}(t)$ is a vector containing the complex signal envelopes of L narrowband signal sources located in the x-y-plane, $\mathbf{n}(t)$ is a $N\times 1$ vector of zero-mean spatially white sensor noise of variance $\sigma_{\mathbf{n}}^2$ and the columns of the steering matrix $\mathbf{A}(\boldsymbol{\theta})$ are the steering or array response vectors $\mathbf{a}(\theta_l)$ corresponding to the unknown source DOAs $\theta_1, \ldots, \theta_L$. The steering vector associated with direction θ is given by

$$\mathbf{a}(\theta) = \left[e^{jk(x_1\cos\theta + y_1\sin\theta)}, \ldots, e^{jk(x_N\cos\theta + y_N\sin\theta)}\right]^T, \tag{2}$$

where $(\cdot)^T$ denotes transposition and k is the wavenumber. The covariance matrix corresponding to (1) is

$$\mathbf{R} = \mathrm{E}\left\{\mathbf{x}(t)\mathbf{x}^H(t)\right\} = \mathbf{A}(\boldsymbol{\theta})\,\mathbf{S}\,\mathbf{A}^H(\boldsymbol{\theta}) + \sigma_{\mathbf{n}}^2\mathbf{I} \tag{3}$$

where $\mathbf{S} = \mathrm{E}\left\{\mathbf{s}(t)\mathbf{s}^H(t)\right\}$ is the source covariance matrix.

3. ARRAY INTERPOLATION PRINCIPLE

In array interpolation, the real array manifold is linearly transformed onto a preliminary specified virtual array manifold over a given angular sector Θ. That is, an interpolation matrix $\mathbf{B}$ is designed in order to satisfy

$$\mathbf{B}\mathbf{a}(\theta) \approx \tilde{\mathbf{a}}(\theta) \quad \forall \quad \theta \in \Theta, \tag{4}$$

where $\mathbf{a}(\theta)$ and $\tilde{\mathbf{a}}(\theta)$ are the $N\times 1$ and $\tilde{N}\times 1$ steering vectors of the real and virtual array, respectively, and $\tilde{N}$ is the number of virtual sensors. The virtual array manifold $\tilde{\mathbf{a}}(\theta)$ usually corresponds to a uniform linear array.

The computation of the interpolation matrix $\mathbf{B}$ is done by choosing K representative directions $\theta_1, \ldots, \theta_K$ from the interpolation sector and minimizing the sum of the quadratic interpolation errors in these directions,

$$\mathrm{F}(\mathbf{B}) = \sum_{i=1}^{K} \left\| \mathbf{B}\,\mathbf{a}(\theta_i) - \tilde{\mathbf{a}}(\theta_i) \right\|^2 = \left\| \mathbf{B}\mathbf{C} - \tilde{\mathbf{C}} \right\|_F^2, \tag{5}$$

with respect to $\mathbf{B}$. In the last equality, we used the matrices $\mathbf{C}$ and $\tilde{\mathbf{C}}$ containing the steering vectors of the real and virtual arrays associated with the K sample directions. The solution of this least-squares optimization is $\mathbf{B} = \tilde{\mathbf{C}}\mathbf{C}^{\mathbf{H}}(\mathbf{C}\mathbf{C}^{\mathbf{H}})^{-\mathbf{1}}$ if the involved inverse exists. Note that this calculation of high computational effort has to be done only once in the design phase, not in the "online" phase as proposed in [6]. In the algorithm application, the incoming data vectors are left-multiplied by the interpolation matrix as $\tilde{\mathbf{x}}(t) = \mathbf{B}\mathbf{x}(t)$. By using a noise-prewhitening matrix, the well-known root-MUSIC algorithm [1] can be applied to the transformed data. For more details, refer to [4].

4. VIRTUAL ARRAY DESIGN ISSUES

In the following, we give some deeper understanding of problems arising in the design of the virtual array geometry. With aid of simulation results, we explain the interrelation between the interpolation error defined in (5), the condition number of the interpolation matrix and the resulting DOA estimation bias. Especially the condition number of the interpolation matrix is very hard to describe in terms of the array geometries. Instead of going deep into the theory, we give visual statements and motivation.

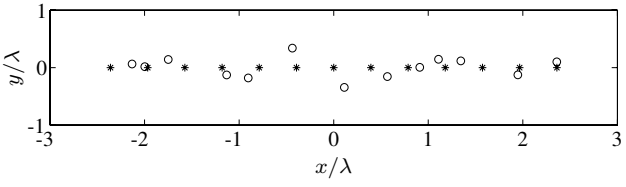

Fig. 1. Geometry of original array and virtual ULA

For the simulation, we used the array geometry depicted in figure 1. The circles show the positions of the $N = 13$ original sensors (with the center of gravity of the sensor locations given at the origin), while the stars show the position of a virtual ULA ($\tilde{N} = N$) used for comparison. The directions-of-arrival of three equipowered signal sources (DOAs fixed to 26.1°, 77.9° and 130.3°) were estimated using the original interpolation method with the ULA as virtual array over the large sector $\Theta = [0°, 180°]$. In several simulation runs, we changed the sensor spacing according to $d = \eta d_0$, where d_0 is the sensor spacing of the virtual ULA shown in the figure and η is a parameter to be varied between 0.85 and 1.15. In each simulation run, we calculated the DOA estimation bias, i.e. the RMSE with no sensor noise being present. Further, we calculated the interpolation error according to (5).

From numerous simulations with Friedlander's interpolation approach, it becomes apparent that a small interpolation error itself does not guarantee a low bias in the DOA estimation. If the condition number is high - which means that the interpolation matrix is close to being singular - the DOA estimation gives poor results. Only if the interpolation matrices calculated by different approaches have comparable condition numbers, the interpolation error can be used to conclude on the expected DOA estimation bias. In figure 2, the interpolation error, condition number and bias are shown over the parameter η. The meaning of the dashed lines will be discussed later.

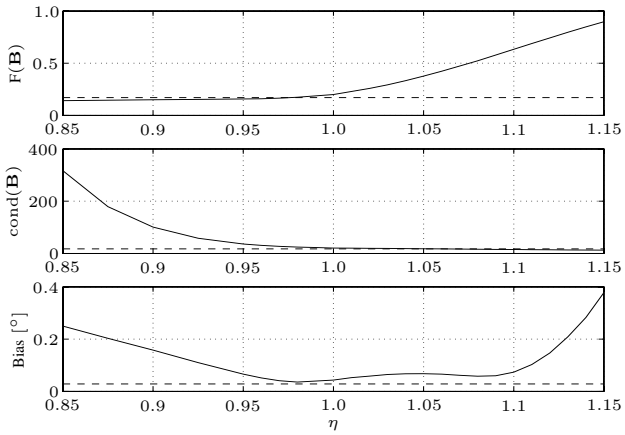

Fig. 2. Interpolation error, condition number and bias

Next, we explain the situation when the inter-element spacing is less or equal to the starting value d_0 (i.e. $0.85 \leq \eta \leq 1$). Let us regard the array interpolation - by its original meaning - as a simple interpolation problem. Looking at a single virtual sensor, the given interpolation task is to approximate a complex function in the parameter θ (i.e., a function of the form $e^{jk(x\cos\theta + y\sin\theta)}$) by a linear combination of a set of complex "basis"-functions, i.e. the elements of the original steering vector (2). If the virtual sensor element is lying very close (in space) to one real sensor, the complex functions of both elements will also be very similar. The approximation will mainly rely on this element and only marginally on the other elements. When the aperture of the virtual array gets smaller, all virtual sensors will be close to some of the inner sensors of the real array. The approximation can be done very well by a linear combination depending on those sensors. If the sensor spacing of the virtual ULA is decreased, the real sensors being closest to a specific virtual sensor will change, but the interpolation error can be expected not to change too much. This is validated by the uppermost graph of figure 2 showing the interpolation error.

On the other hand, the outer, more distant sensors of the real array are less useful for the approximation, because the complex functions of the virtual sensors and the real sensors differ significantly. The linear factors associated with these sensors will be very small. As it can easily be verified by (4), each row of the interpolation matrix contains the linear factors for the interpolation of one virtual sensor, and, in turn, each column of the interpola-

tion matrix contains the interpolation factors associated with one of the real sensors. When very small values for the interpolation are associated with some of the sensors of the real array according to all of the virtual sensors, the interpolation matrix will contain columns with very small elements. This, however, means that the interpolation matrix will be close to singular. In the center graph of figure 2 this claim is validated by the significantly rising condition number at decreasing sensor spacing.

An opposite situation occurs when increasing the inter-element spacing starting from $d = d_0$, i.e., increasing η from 1 up to 1.15. Now, the distances between the outermost virtual sensors and any real sensor are getting larger. The "basis"-functions differ more and more from the functions of the virtual sensors, so the approximation quality will clearly get worse, leading to a high interpolation error. In contrast to the former situation with low virtual aperture, where it was quite clear which of the real sensors are most useful for the approximation, here the linear factors are nearly unpredictable. The interpolation matrix will contain "randomly distributed" numbers resulting in a good conditioning. These facts are also approved by the simulation results in figure 2.

As a first conclusion, the lower of the three graphs affirms our statement that both the interpolation error and the condition number of the interpolation matrix account for the DOA estimation bias. The graph looks in a way like the product of the other two graphs, as there is a minimum in the middle and the bias is rising at the edges.

Next, we will introduce the arc length of a sensor array and use it to derive a design method that will lead to both a small interpolation error and a well-conditioned interpolation matrix.

5. ARC LENGTH AND VIRTUAL ARRAY DESIGN

The complex steering vector $\mathbf{a}(\theta)$ of a sensor array is usually parameterized by the angle θ and describes a curve in the N-dimensional complex space. As we can calculate the traveled way of a moving object as the integral of its speed over the time, we can calculate the arc length $s(\theta)$ of the curve by integrating the rate of change of the steering vector (see [7]), i.e.

$$s(\theta) = \int_0^\theta \dot{s}(\vartheta) d\vartheta = \int_0^\theta \left\| \frac{d\mathbf{a}(\vartheta)}{d\vartheta} \right\| d\vartheta. \tag{6}$$

The arc length can easily be calculated in the case of a uniform linear array on the x-axis. Here, the nth element ($n = 0, \ldots, N-1$) of the steering vector is

$$[\mathbf{a}(\theta)]_n = e^{jnkd\cos(\theta)}\ , \tag{7}$$

its derivative is

$$\frac{d}{d\theta}[\mathbf{a}(\theta)]_n = -jnkd\sin(\theta)e^{jnkd\cos(\theta)}. \tag{8}$$

The resulting norm of the derivative of the steering vector is

$$\dot{s}(\theta) = \mu kd|\sin(\theta)| \quad \text{with} \quad \mu^2 = \sum_{n=0}^{N-1} n^2. \tag{9}$$

This expression shows that two closely spaced sources impinging on the array from broadside differ significantly more in their corresponding steering vectors than two closely spaced sources in the endfire directions. This is one way of justification for the well-known fact that the accuracy of DOA estimation is best in the main direction of an ULA. Finally, the arc length of the uniform linear array can be calculated with (6):

$$s(\theta) = \mu kd(1 - \cos(\theta))\ , \quad \theta \in [0^\circ, 180^\circ] \tag{10}$$

When a circular-like real array and a virtual ULA are used, the rate of change of the real steering vectors is nearly constant, while for the virtual sensors the rate varies in a sinusoidal manner. This change in behavior can not be satisfactory accounted for by a simple linear transformation and thus gives poor results especially over a large interpolation sector.

Our intention now is to find a virtual array manifold that has a Vandermonde structure to be exploited by the root-MUSIC algorithm and the same directional behavior as the original array. The first condition can easily be fulfilled by constructing the virtual steering vector as

$$\tilde{\mathbf{a}}(\theta) = \left[1, e^{jg(\theta)}, \ldots, e^{j(\tilde{N}-1)g(\theta)}\right]^T \tag{11}$$

with an invertible function $g(\theta)$ to be defined in the following. As a measure for the directional behavior, we use the rate of change of the arc length. For this newly defined virtual steering vector, it is

$$\dot{\tilde{s}}(\theta) = \left\| \frac{d\tilde{\mathbf{a}}(\vartheta)}{d\vartheta} \right\| = \tilde{\mu}|\dot{g}(\theta)| \quad \text{with} \quad \tilde{\mu}^2 = \sum_{n=0}^{\tilde{N}-1} n^2. \tag{12}$$

Obviously, the function

$$g(\theta) = \frac{1}{\tilde{\mu}} \int_0^\theta \left\| \frac{d\mathbf{a}(\vartheta)}{d\vartheta} \right\| d\vartheta \tag{13}$$

satisfies the second condition $\dot{s}(\theta) = \dot{\tilde{s}}(\theta)$ (see (6)).

In opposite to the original array interpolation approaches, where in some cases the number of virtual sensors has to be reduced in order to get a reasonably conditioned interpolation matrix [4], we expect that with our design method this is not necessary at all. However, if desired, the virtual array designer still has the option to choose a smaller number of virtual sensors in order to further reduce the computational effort.

Using the virtual steering vector constructed in (11) instead of the steering vector of a ULA, the original array interpolation procedure with the root-MUSIC algorithm can still be applied. The only difference is that the DOA estimations $\hat{\theta}_n$ have to be calculated from the roots $\hat{z}_n$ of the root-MUSIC-polynomial by the inversion formula

$$\hat{\theta}_n = g^{-1}(\arg(\hat{z}_n)) \tag{14}$$

instead of $\hat{\theta}_n = \arcsin(\arg(\hat{z}_n)/kd)$.

While this function can always be easily calculated analytically, it is generally not possible to invert it analytically like in the ULA case. For this reason, representing the function $g^{-1}(\theta)$ by second-order splines is a reasonable approach. The spline interpolation error is deterministic and directly affects the DOA estimation. It can be kept negligible in comparison to the stochastic error caused by the sensor noise by selecting appropriately small intervals for the spline representation.

6. SIMULATION RESULTS

In the following, we will illustrate the advantages of our new design method by the results of a second simulation using the same

array geometry as before (figure 1). In this simulation, in each of 1000 runs three source positions were chosen randomly. The DOAs were estimated at the presence of Gaussian noise with an SNR of 0 dB. First, Friedlander's original interpolation approach was applied with the sensor spacing d_0 of the virtual ULA. Second, we used our approach with the modified virtual array manifold based on the arc length of the original array. In fact, due to the centering of the original array to the origin, instead of (11) the virtual array manifold

$$\tilde{\mathbf{a}}(\theta) = \left[e^{-j\frac{\tilde{N}-1}{2}g(\theta)}, \ldots, 1, \ldots, e^{j\frac{\tilde{N}-1}{2}g(\theta)}\right] \qquad (15)$$

was used (with $\tilde{\mu}$ appropriately modified). The interpolation sector was again $\Theta = [0^\circ, 180^\circ]$.

The result shown in figure 3 was calculated as follows: First all DOAs falling into intervals of 10° width (e.g. $[30^\circ, 40^\circ)$) were collected. Then the RMSE of the corresponding DOA estimations was calculated and gives the value at the interval center (e.g. at 35°). It is remarkable that there is no difference in RMSE over a large interval of the interpolation sector. Only close to the endfire directions, our proposed design method shows little advantage over the common interpolation approach. However, in the RMSE calculation all DOA estimates with an error larger than 20° were omitted. With the virtual ULA, about 60 percent of the DOA estimations for real DOAs lying in the intervals $[0^\circ, 10^\circ)$ and $[170^\circ, 180^\circ]$ were outliers. Using our proposed design method, we found that there were less than 10 percent outliers. This result shows that our proposed design method allows the use of very large interpolation sectors, as the DOA estimation works well even close to the endfire directions.

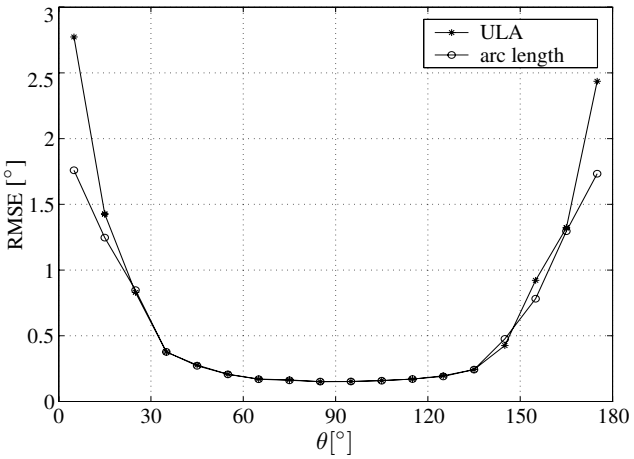

Fig. 3. RMSE at SNR=0 dB

It has not been discussed yet, how the parameters of the virtual ULA have been chosen, which - as was said before - is usually done heuristically. The optimal center position of the ULA was quite clear, as we placed the center of gravity of the sensor positions of both the original and the virtual ULA to the origin. The inter-element spacing, on the other hand, could still have been chosen slightly larger or smaller without hurting the condition that the apertures of both arrays shall be of similar size. For this purpose, we chose the inter-element spacing just in a way that the arc length of original array and virtual ULA fitted best in a least-squares-sense. This can easily be done, as the sensor spacing is a linear factor in the arc length, see equation (10). Clearly, the RMSE values attained with the two different approaches do not differ very much, because the steering vectors of both virtual arrays are very similar to each other.

The former simulation presented in chapter 4 was once done using the proposed virtual array manifold instead of the virtual ULA as the interpolation target. The results (of course independent of η) are shown in figure 2 by the dashed lines. Our method of designing the virtual array manifold gains good values in the interpolation error and the condition number and outscores the original method by Friedlander in DOA estimation bias even in the case where the parameters of the virtual ULA are chosen nearly optimal.

7. CONCLUSION

In this paper, we first examine the interrelations between interpolation error, condition number of the interpolation matrix and the resulting DOA estimation bias. Motivated by the changing resolution capabilities of sensor arrays in different directions-of-arrival, we propose a new method of designing the virtual array manifold based on the arc length representation of the real sensor array. By construction, it matches the directional behavior of the real array. Only minor changes to the original DOA estimation procedure using a virtual ULA and the root-MUSIC algorithm are necessary. In contrast to common approaches, this design method does not require any of the properties of the virtual array geometry to be chosen heuristically. Further, it provides the option to reduce the dimensionality of the data by choosing a number of virtual sensors less than the number of real sensors. Simulation results show that our method outperforms Friedlander's original approach and allows the use of very large interpolation sectors.

8. REFERENCES

[1] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," *ICASSP '83*, Boston, Massachusetts, pp. 336-339, May 1983.

[2] M. Pesavento, A. B. Gershman and K. M. Wong, "Direction of arrival estimation in partly calibrated time-varying sensor arrays," *Proceedings ICASSP'01*, Salt Lake City, Utah, vol. 5, pp. 3005-3008, May 2001.

[3] A. L. Swindlehurst, B. Ottersten, R. Roy and T. Kailath, "Multiple invariance ESPRIT," *IEEE Transactions on Signal Processing*, vol. 40, pp. 867-881, April 1992.

[4] B. Friedlander, "The root-MUSIC algorithm for direction finding with interpolated arrays," *Signal Processing*, vol. 30, pp. 15-29, 1993.

[5] M. Bühren, M. Pesavento, J. F. Böhme, "A new approach to array interpolation by generation of artificial shift invariances: Interpolated ESPRIT," *Proc. ICASSP'03*, Hong Kong, vol. 5, pp. 205-208, April 2003.

[6] P. Hyberg, M. Jansson and B. Ottersten, "Array mapping: optimal transformation matrix design," *Proc. ICASSP'02*, Orlando, Florida, pp. 2905-2908, May 2002.

[7] A. Manikas, N. H. Dowlut, "The Use of Differential Geometry in Array Signal Processing," *Kybernetes*, vol. 27, no. 3, pp. 251-263, 1998.