

Article

Constant-Beamwidth LCMV Beamformer with Rectangular Arrays

Vitor Probst Curtarelli ^{1,*}  and Israel Cohen ² 

¹ Graduate Program in Electrical Engineering, Universidade Federal de Santa Catarina, Florianópolis CEP 88040-900, SC, Brazil

² Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion–Israel Institute of Technology, Technion City, Haifa 3200003, Israel; icohen@ee.technion.ac.il

* Correspondence: vitor.curtarelli@gmail.com

Abstract: This paper presents a novel approach utilizing uniform rectangular arrays to design a constant-beamwidth (CB) linearly constrained minimum variance (LCMV) beamformer, which also improves white noise gain and directivity. By employing a generalization of the convolutional Kronecker product beamforming technique, we decompose a physical array into virtual subarrays, each tailored to achieve a specific desired feature, and we subsequently synthesize the original array's beamformer. Through simulations, we demonstrate that the proposed approach successfully achieves the desired beamforming characteristics while maintaining favorable levels of white noise gain and directivity. A comparative analysis against existing methods from the literature reveals that the proposed method performs better than the existing methods.

Keywords: LCMV beamformer; constant-beamwidth beamforming; Kronecker product beamformer; array signal processing; rectangular sensor arrays

1. Introduction

Beamformers play a crucial role in diverse fields, such as telecommunications [1], acoustics [2,3], hearing aids [4], and others [5–8]. Among the array configurations used for beamforming, rectangular arrays are an interesting option to be explored [9–11], as they offer distinct advantages over linear arrays, providing enhanced spatial information regarding impinging sources [12,13] and reduced redundancy due to their asymmetry [14].

The development of robust adaptive beamformers with frequency-invariant characteristics has been a significant point of interest as, in this case, frequency does not affect the behavior of the beamformer. Some desired features are a constant beamwidth [15] and null steering [16]. One approach for null steering is using a linearly constrained minimum variance (LCMV) beamformer [17–19], which cancels interfering signals from given directions and steers the main beam toward the desired signal. However, it lacks a robust mechanism for maintaining a constant beamwidth. Constant-beamwidth (CB) beamforming [15,20,21] can be accomplished by using window-based beamforming techniques [22], but these methods cannot incorporate directional restrictions. CB-LCMV beamformers have been recently explored [23], however only in the context of linear sensor arrays, leaving space for their study in the context of different array configurations.

Other metrics that are desired to be enhanced are white noise gain and directivity factor, which can respectively be maximized by the delay-and-sum (DS) and superdirective (SD) beamformers [24]. Another quality that is often required for designing a beamformer is a distortionless response to the desired source or to the desired source direction. This ensures that the desired signal is unaltered by filtering processes.

While constructing a beamformer with multiple beamforming features is non-trivial, efforts have been made to combine multiple beamforming techniques for a single array of sensors. Two notable approaches are the Kronecker product (KP) method [25,26] and the



Citation: Curtarelli, V. P.; Cohen, I. Constant-Beamwidth LCMV Beamformer with Rectangular Arrays. *Algorithms* **2023**, *16*, 385. <https://doi.org/10.3390/a16080385>

Academic Editor: Frank Werner

Received: 12 July 2023

Revised: 27 July 2023

Accepted: 28 July 2023

Published: 10 August 2023



Copyright: © 2023 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 (<https://creativecommons.org/licenses/by/4.0/>).

linear convolutional Kronecker product (LCKP) method [23]. The LCKP method is valid only for linear arrays, however it allows for the virtual utilization of more sensors than are physically available [23]. Meanwhile, the KP method can be used in linear or rectangular arrays, but it does not increase the number of sensors available for beamforming. For both combination methods (the KP and LCKP), beam pattern features and distortionless constraints are preserved in the combined beamformers. Therefore, by using beamformers with desired beam pattern characteristics that respect the distortionless constraint, these are maintained in the end result.

In this paper, we propose a novel approach to constructing a CB-LCMV beamformer for rectangular arrays. To this end, we generalize the LCKP beamforming technique to the case of rectangular arrays. We synthesize beamformers for virtual subarrays and, with our proposed generalized technique, apply them to a full array to achieve the desired beamwidth and null placement. This is achieved without sacrificing white noise gain or the directivity factor. The performance of the proposed method is compared against that of beamformers obtained through the KP and LCKP methods. Our results demonstrate superior performance in terms of the beamwidth, white noise gain, and directivity of the beamformers obtained using the proposed method when compared to the performance of the beamformers obtained using the methods in the literature.

This paper is organized as follows: Section 2 presents the array and signal model considered for the problem; Section 3 shows the traditional beamforming techniques and methods that are used further. In Section 4, the newly proposed method for an array analysis for rectangular arrays is introduced and detailed, as well as its proposed usage for solving the problem at hand. In Section 5, we present the conducted simulations and discuss the results, comparing them to those in the literature. Finally, Section 6 concludes this paper, presenting an overview of the main contributions.

2. Signal and Array Model

Let S be a uniform rectangular array (URA) of sensors over the $x - y$ plane in an anechoic environment with desired and undesired sources. The URA comprises M_x sensors spaced δ_x apart along the x -axis and M_y sensors spaced δ_y apart along the y -axis, resulting in a total of $M = M_x M_y$ sensors. Assume a source in the far field on the same plane as the sensor array (that is, with elevation $\phi = 0^\circ$), impinging on it from an azimuth angle θ . As it is unusual, in speech enhancement, for the desired and undesired sources to have the same azimuth, differing only by elevation, we assume the elevation to be 0° . This constraint can be easily removed without affecting the developed mathematical framework.

Let $\mathbf{D}(\omega, \theta)$ denote the steering matrix of size $M_x \times M_y$ with elements $\{m_x, m_y\}$ given by

$$[\mathbf{D}(\omega, \theta)]_{m_x, m_y} = \exp\left\{-j\frac{\omega}{c} r_{m_x, m_y} \cos(\theta - \psi_{m_x, m_y})\right\}, \quad (1)$$

where $c = 340$ m/s is the speed of sound, $(r_{m_x, m_y}, \psi_{m_x, m_y})$ are the polar coordinates of the sensor at $(m_x \delta_x, m_y \delta_y)$, $\omega = 2\pi f$ is the angular frequency, f is the temporal frequency, and $j = \sqrt{-1}$ is the imaginary unit. We let $\mathbf{d}(\omega, \theta) = \mathcal{V}(\mathbf{D}(\omega, \theta))$ denote the $M \times 1$ steering vector, with $M = m_x m_y$ and $\mathcal{V}(\cdot)$ being the vectorization operation, and we let $\mathbf{D}(\omega, \theta) = \mathcal{V}^{-1}(\mathbf{d}(\omega, \theta); M_y)$ denote the inverse vectorization of $\mathbf{d}(\omega, \theta)$.

The $M \times 1$ observed signal vector $\mathbf{y}(\omega)$ for all sensors in the frequency domain can be written as

$$\mathbf{y}(\omega) = \mathbf{d}(\omega, \theta_d) X(\omega) + \mathbf{v}(\omega), \quad (2)$$

where $X(\omega)$ is the desired signal at the reference sensor, $\mathbf{d}(\omega, \theta_d)$ is the steering vector of the desired source from direction θ_d , and $\mathbf{v}(\omega)$ is the additive noise signal vector. All signals are assumed to be zero-mean and uncorrelated. We can estimate the desired signal $X(\omega)$ as $Z(\omega)$ using the beamformer $\mathbf{h}(\omega)$ (assumed to be a 2-D beamformer) through linear filtering,

$$Z(\omega) = \mathbf{h}(\omega)^H \mathbf{y}(\omega), \quad (3)$$

where the superscript H denotes the conjugate-transpose operator. In that case,

$$Z(\omega) = X(\omega) + \mathbf{v}_{\text{rn}}(\omega), \tag{4}$$

where $\mathbf{v}_{\text{rn}}(\omega) = \mathbf{h}(\omega)^{\text{H}}\mathbf{v}(\omega)$ is the residual noise at the beamformer’s output. The beamformer is called distortionless if it satisfies $\mathbf{h}(\omega)^{\text{H}}\mathbf{d}(\omega, \theta_{\text{d}}) = 1$ for all ω . This constraint guarantees that the beamformer does not affect the desired signal, only altering the undesired noise signal.

From here on, ω is omitted unless in definitions, and θ in the steering vectors appears in subscripts where necessary. When no angle is shown, $\mathbf{d} = \mathbf{d}(\omega, \theta_{\text{d}})$ is assumed to be the desired signal steering vector for conciseness.

Beamformer Metrics

Beampattern \mathcal{B} , as a function of beamformer \mathbf{h} and direction θ (through steering vector \mathbf{d}_{θ}) is given by

$$\mathcal{B}(\mathbf{h}, \mathbf{d}_{\theta}) = \mathbf{h}^{\text{H}}\mathbf{d}_{\theta}. \tag{5}$$

Given the desired signal steering vector \mathbf{d} , the white noise gain (WNG), desired signal distortion index (DSDI), and directivity factor (DF) are, respectively,

$$\mathcal{W}(\mathbf{h}, \mathbf{d}) = \frac{|\mathbf{h}^{\text{H}}\mathbf{d}|^2}{\mathbf{h}^{\text{H}}\mathbf{h}}, \tag{6a}$$

$$v_{\text{d}}(\mathbf{h}, \mathbf{d}) = \left| \mathbf{h}^{\text{H}}\mathbf{d} - 1 \right|^2, \tag{6b}$$

$$\mathcal{D}(\mathbf{h}, \mathbf{d}) = \frac{|\mathbf{h}^{\text{H}}\mathbf{d}|^2}{\mathbf{h}^{\text{H}}\mathbf{\Gamma}\mathbf{h}}, \tag{6c}$$

where $\mathbf{\Gamma}(\omega)$ is the spherical isotropic noise field coherence matrix [27]. Using the DSDI, the distortionless constraint can also be written as $v_{\text{d}}(\mathbf{h}, \mathbf{d}) = 0$.

3. Conventional Beamformers

This section briefly overviews the beamforming techniques used to construct the proposed beamformer, as well as the different methods for beamformer synthesis.

3.1. LCMV Beamformer

Assuming the presence of undesired sources in known directions, a linearly constrained minimum variance (LCMV) beamformer [17–19] is useful to position the nulls of the beampattern in those undesired directions. To this end, we assume the existence of N (with $N < M$) uncorrelated interfering sources in the far field, each coming from a (different) direction θ_i ($i \in \{ 1, \dots, N \}$) that we want to cancel. We write \mathbf{v} as

$$\mathbf{v} = \sum_{n=1}^N \mathbf{d}_{\theta_n} v_n + \mathbf{u}, \tag{7}$$

where v_n is the noise signal for the n -th undesired direction, and \mathbf{u} is the portion of the noise signal not coming from the N undesired directions, also accounting for acoustic uncorrelated noise. We assume that $\text{E} \left[\mathbf{u}^{\text{H}}(\mathbf{d}_{\theta_n} v_n) \right] = 0$. We then use $N + 1$ linear constraints, representing the distortionless constraint for the desired signal plus the canceling of the N undesired directions.

The LCMV constraint is written in matrix form as

$$\mathbf{C} = \left[\mathbf{d}, \mathbf{d}_{\theta_1}, \dots, \mathbf{d}_{\theta_N} \right], \tag{8a}$$

$$\mathbf{q} = \left[1, 0, \dots, 0 \right]^{\text{T}}, \tag{8b}$$

$$\mathbf{C}^H \mathbf{h} = \mathbf{q}, \tag{8c}$$

where $\mathbf{C}(\omega)$ is an $M \times (N + 1)$ matrix, \mathbf{q} is an $(N + 1) \times 1$ vector, and the superscript T denotes the transpose operator. The LCMV beamformer is obtained by minimizing the variance of residual noise \mathbf{v}_m in the beamformer output (from Equation (4)), given the constraints in (8), which translates to a minimization problem given as

$$\mathbf{h}_{\text{LCMV}} = \underset{\mathbf{h}}{\text{argmin}} \mathbf{h}^H \Phi_{\mathbf{v}} \mathbf{h} \text{ s.t. } \mathbf{C}^H \mathbf{h} = \mathbf{q}, \tag{9}$$

where $\Phi_{\mathbf{v}}(\omega) = \text{E}[\mathbf{v}(\omega)\mathbf{v}^H(\omega)]$ is the correlation matrix of \mathbf{v} , assumed to be a full-rank invertible matrix. As the undesired directions will be canceled (assuming an anechoic environment [28]), this minimization process minimizes \mathbf{u} , the noise portion that is uncorrelated with the N undesired directions. The solution to this minimization problem is

$$\mathbf{h}_{\text{LCMV}} = \Phi_{\mathbf{v}}^{-1} \mathbf{C} [\mathbf{C}^H \Phi_{\mathbf{v}}^{-1} \mathbf{C}]^{-1} \mathbf{q}. \tag{10}$$

To ensure the existence of a solution, the number of sensors should be larger than or equal to the number of constraints, i.e., $M \geq N + 1$. For $N = 0$, only the distortionless constraint remains, and the LCMV beamformer reduces to a minimum variance distortionless response (MVDR) beamformer [29], which is given by

$$\mathbf{h}_{\text{MVDR}} = \frac{\Phi_{\mathbf{v}}^{-1} \mathbf{d}}{\mathbf{d}^H \Phi_{\mathbf{v}}^{-1} \mathbf{d}}. \tag{11}$$

It is possible to show that the LCMV and MVDR beamformers are also defined in terms of the observed signal correlation matrix $\Phi_{\mathbf{y}}$ [24], in which case

$$\mathbf{h}_{\text{LCMV}} = \Phi_{\mathbf{y}}^{-1} \mathbf{C} [\mathbf{C}^H \Phi_{\mathbf{y}}^{-1} \mathbf{C}]^{-1} \mathbf{q} \tag{12a}$$

$$\mathbf{h}_{\text{MVDR}} = \frac{\Phi_{\mathbf{y}}^{-1} \mathbf{d}}{\mathbf{d}^H \Phi_{\mathbf{y}}^{-1} \mathbf{d}}. \tag{12b}$$

This formulation depends only on the statistics of the observed signal, which are easier to compute than those of the noise signal.

3.2. CB Beamformer

Constant-beamwidth (CB) beamformers guarantee a certain beamwidth around the desired direction that is constant over frequency. This is important to ensure the correct receiving of the desired signal, even if θ_d is not precisely calibrated.

We define θ_B as the first-null beamwidth (FNBW), such that $|\mathcal{B}(\mathbf{h}, \mathbf{d})| > 0$ if $|\theta - \theta_d| < \theta_B / 2$. That is, $\theta_B / 2$ is the first angle in which a null of the beampattern occurs. A constant-beamwidth beamformer can be achieved using a window-based design technique [22]. Here, the Kaiser window is used [30], which can be written as

$$[\mathbf{w}]_m = \frac{J_0\left(\beta \sqrt{1 - \left[\frac{2m}{M-1} - 1\right]^2}\right)}{J_0(\beta)}, \tag{13}$$

where $J_0(\cdot)$ is the zero-order modified Bessel function of the first kind, and $\beta(\omega)$ ([22], Section 3.3.2) is frequency-dependent to maintain the θ_B constant. This technique requires that the desired source signal impinges on the array from the broadside direction [22]. To satisfy the distortionless constraint, we normalize \mathbf{w} , obtaining

$$\mathbf{h}_{\text{CB}} = \frac{\mathbf{w}}{\sum_{m=0}^{M-1} [\mathbf{w}]_m}. \tag{14}$$

3.3. SD and DS Beamformers

Superdirective (SD) and delay-and-sum (DS) beamformers are obtained by maximizing the DF and the WNG, respectively [29,31], both subject to the distortionless constraint. The solutions to these minimization problems are respectively given by

$$\mathbf{h}_{SD} = \frac{\mathbf{G}^{-1}\mathbf{d}}{\mathbf{d}^H\mathbf{G}^{-1}\mathbf{d}}, \tag{15a}$$

$$\mathbf{h}_{DS} = \frac{\mathbf{d}}{M}. \tag{15b}$$

3.4. Kronecker Product Beamforming

Designing a single beamformer with different features is highly important, allowing different effects to be employed over a single array of sensors. Two methods to accomplish such a task are the Kronecker product (KP) [25,26] and linear convolutional Kronecker product (LCKP) methods [23]. In these processes, the sensor array is split into subarrays for which we design separate beamformers and use the chosen technique to synthesize the whole sensor array's (or full array's) beamformer.

The KP beamforming process is as follows: given steering vector \mathbf{d}_θ , we decompose it into two parts (namely, $\mathbf{d}_{1,\theta}$ and $\mathbf{d}_{2,\theta}$) satisfying the relation $\mathbf{d}_\theta = \mathbf{d}_{1,\theta} \otimes \mathbf{d}_{2,\theta}$, where \otimes represents the Kronecker product. By designing beamformers \mathbf{h}_1 and \mathbf{h}_2 for \mathbf{d}_1 and \mathbf{d}_2 , respectively, we obtain the beamformer for the full array as $\mathbf{h} = \mathbf{h}_1 \otimes \mathbf{h}_2$ [32].

LCKP beamforming is achieved similarly: given a uniform linear array's (ULA) steering vector \mathbf{d}_θ of length M , we define $\mathbf{d}_{1,\theta}$ as the M_1 -th first elements of \mathbf{d}_θ and, similarly, $\mathbf{d}_{2,\theta}$ with M_2 elements, respecting $M_1 + M_2 - 1 = M$. By designing beamformers \mathbf{h}_1 and \mathbf{h}_2 for each subarray, the full array's beamformer is $\mathbf{h} = \mathbf{h}_1 * \mathbf{h}_2$ [23], where $*$ denotes the linear convolution.

For both the KP and LCKP methods, we have the following properties:

$$\mathcal{B}(\mathbf{h}, \mathbf{d}_\theta) = \mathcal{B}(\mathbf{h}_1, \mathbf{d}_{1,\theta}) \mathcal{B}(\mathbf{h}_2, \mathbf{d}_{2,\theta}), \tag{16a}$$

$$v_d(\mathbf{h}, \mathbf{d}) \leq [1 + v_d(\mathbf{h}_1, \mathbf{d}_1)][1 + v_d(\mathbf{h}_2, \mathbf{d}_2)] - 1. \tag{16b}$$

The first one shows that the beampattern of \mathbf{h} is the combination of the beampatterns of the subarrays. Through the second one, we can see that, if \mathbf{h}_1 and \mathbf{h}_2 are distortionless beamformers, \mathbf{h} also will be. From these properties, we can see that the beampattern properties (including the distortionless feature) from \mathbf{h}_1 and \mathbf{h}_2 are maintained for \mathbf{h} .

4. Constant-Beamwidth LCMV Beamformer with Rectangular Arrays

Both the KP and LCKP beamforming methods have advantages and disadvantages. While the LCKP is only usable over linear arrays, beamformers achieved through it have (virtually) more sensors than there are available in the physical array, generally leading to better performance when combining different techniques. Meanwhile, the KP method can be applied to rectangular arrays, which on its own is beneficial, but it does not have the virtual utilization of more sensors. To benefit from the lack of symmetry of the rectangular arrays and to also be able to have more sensors available for each subarray's beamformer, we propose a generalization of the LCKP to URAs to take advantage of both methods, enabling virtual sensor augmentation while exploiting the rectangular array's symmetry.

4.1. Rectangular Convolutional Kronecker Product Beamforming

Let S_1 be a subarray of S , including the $M_{1,x} \times M_{1,y}$ first sensors of S , with steering vector \mathbf{d}_1 , and similarly for S_2 . These arrays sizes are such that $M_{1,x} + M_{2,x} - 1 = M_x$, and $M_{1,y} + M_{2,y} - 1 = M_y$. By designing beamformers \mathbf{h}_1 and \mathbf{h}_2 for S_1 and S_2 , respectively, we show in Appendix A that it is possible to synthesize a beamformer for full array S through

$$\mathbf{h} = \mathcal{V} \left(\mathcal{V}^{-1}(\mathbf{h}_1; M_{1,y}) \circledast \mathcal{V}^{-1}(\mathbf{h}_2; M_{2,y}) \right), \tag{17}$$

with \otimes representing the 2-D convolution operation. We call this the rectangular convolutional Kronecker product (RCKP) method. A simple implementation of the proposed RCKP method is presented in Algorithm A1 in Appendix B, which is written in a Python-like pseudolanguage.

In the same way as with the KP and LCKP methods, with the RCKP, one can design beamformers for subarrays S_1 and S_2 , and one can synthesize the URA’s beamformer \mathbf{h} through the 2-D convolution (and vectorization processes). Accordingly, \mathbf{h} is an $M \times 1$ vector; however, in this scenario we have a total of $M' = M_{1,x}M_{1,y}M_{2,x}M_{2,y}$ virtual sensors. Assuming that all Ms are ≥ 1 , it is trivial to see that $M' \geq M$ (equality happens if S_1 and S_2 are perpendicular ULAs, or if one of them has only one sensor). Therefore, we have more sensors virtually than if S was split into two VAs through the KP method.

It is easy to verify that the properties in (16) are also valid for the proposed RCKP. With this, like for the KP and LCKP, the full-array beamformer obtained through the RCKP inherits the beampattern and distortionless features from the subarray beamformers. Also, since convolution is commutative and associative, one can split the S array into more than two virtual arrays, design a beamformer for each subarray, and combine all the beamformers through the RCKP without the loss of generalization. In this case, assuming that we are using K beamformers, each with size $M_{k,x} \times M_{k,y}$, then their dimensions must be such that

$$\sum_{k=1}^K M_{k,x} - (K - 1) = M_x, \tag{18a}$$

$$\sum_{k=1}^K M_{k,y} - (K - 1) = M_y. \tag{18b}$$

This is easily verifiable by repeating the synthesis operation in Equation (17) $K - 1$ times.

4.2. CB-LCMV Beamformer with RCKP

Here, we propose the use of the RCKP method as derived previously to construct a CB-LCMV beamformer with an increase in white noise gain and directivity measures. For this, full array S is separated into four subarrays: S_1 , $S'_{2,x}$, $S''_{2,x}$, and $S_{2,y}$. Each subarray is used to design one of the desired beamformers presented in Sections 3.1–3.3.

S_1 is used to design the LCMV beamformer, following the steps detailed in Section 3.1. Since the LCMV beamformer does not require the array to be linear, we use S_1 as a rectangular array of size $M_{1,x} \times M_{1,y}$. As explained previously, this choice has the advantage of the rectangular array’s lesser symmetry compared to a linear array. We have $M_1 = M_{1,x}M_{1,y}$ virtual sensors and, at most, $M_1 - 1$ nulls to be placed. If $N = 0$, this same array is used for the MVDR beamformer instead.

The subarray $S_{2,y}$ is used to construct the CB beamformer. Given that the CB is achieved through the window technique (as per Section 3.2), this subarray must be a ULA, and the desired source direction should be on its broadside direction. Here, this implies that $\theta_d = 0^\circ$.

The SD and DS beamformers are built from the $S'_{2,x}$ and $S''_{2,x}$ subarrays, respectively, based on Section 3.3. We assume that they are constructed from linear arrays, but this condition is flexible and can be changed in other implementations.

Once all the subarray beamformers are designed, we use the RCKP method to combine these beamformers into a full-array beamformer. Thus, we construct a beamformer with many desired features (null placement + constant beamwidth + white noise gain + directivity factor gain) that exploit the symmetry of the rectangular array, which implies more spatial information and more performance. Algebraically, the full-array beamformer is given by

$$\mathbf{h} = \mathcal{V}\left(\mathcal{V}^{-1}(\mathbf{h}_{\text{LCMV}}; M_{1,y}) \otimes \mathcal{V}^{-1}(\mathbf{h}_{\text{SD}}; 1) \otimes \mathcal{V}^{-1}(\mathbf{h}_{\text{DS}}; 1) \otimes \mathcal{V}^{-1}(\mathbf{h}_{\text{CB}}; M_{2,y})\right). \tag{19}$$

An implementation of the proposed CB-LCMV beamformer is shown in Algorithm A2 in Appendix B. The procedure calculates beamformer \mathbf{h} for a single frequency.

For the CB beamformer to be effective, its condition must be valid for all beamformers. That is, $|\mathcal{B}(\mathbf{h}, \mathbf{d})| > 0$ if $|\theta - \theta_d| < \theta_B / 2$, where $\mathbf{h} \in [\mathbf{h}_{\text{LCMV}}, \mathbf{h}_{\text{CB}}, \mathbf{h}_{\text{SD}}, \mathbf{h}_{\text{DS}}]$. This is true by definition for the CB beamformer, and by setting $N = M_1 - 1$ for the LCMV, no nulls are free to end up inside the main beam. For the SD and DS beamformers, we assume that their FNBWs are sufficient to satisfy the condition.

5. Experimental Results

In this section, we present the simulations performed to verify the proposed method's implementation and to compare its performance to that of existing methods. The code used for these simulations is available at <https://github.com/VCurtarelli/py-cb-lcmv-rect> (accessed on 27 July 2023). We test different combinations of the number of sensors for each subarray, as established in Table 1. The table presents the dimensions for each subarray, for the full array of sensors, and the virtual number of sensors being employed. [A, B] are obtained through the RCKP, [C, D] through the LCKP, and [E, F] through the KP + LCKP, with the KP and LCKP methods being as defined in Section 3.4. [A, C, E] utilize the SD and DS beamformers, while [B, D, F] "disable" them in favor of the CB beamformer. For [E, F], the LCMV beamformer is more spaced out.

Table 1. Number of sensors for each subarray, dimension of the full array (FA), and size of the virtual array in each simulation.

Cond.	LCMV	SD	DS	CB	FA	M'
A	2×2	2×1	2×1	1×8	4×9	128
B	2×2	1×1	1×1	1×17	2×18	68
C	1×4	1×2	1×2	1×31	1×36	496
D	1×4	1×1	1×1	1×33	1×36	132
E	2×2	2×1	2×1	1×3	6×6	48
F	2×2	1×1	1×1	1×9	2×18	36

In all situations, S has a total of $M = 36$ sensors. The sensors are assumed to be ideal omnidirectional sensors with a plain frequency response over the spectrum. The intersensor distances are $\delta_x = 0.5$ cm and $\delta_y = 3.0$ cm. We assume that $\theta_d = 0^\circ$ and $\theta_B = 40^\circ$. Since the LCMV has four sensors, we use three interfering sources (for all situations), with directions $\theta_i \in [-90^\circ, 60^\circ, 130^\circ]$, each with $\phi_{v_i} = 1$, and we also assume \mathbf{v}' to be uncorrelated Gaussian white noise with unit variance (that is, $\Phi_{\mathbf{v}'}$ is the identity matrix). The variance of the desired signal is $\phi_X = \mathbb{E}[|X|^2] = 5$. The simulations are performed for the range $f \in [4, 8]$ kHz. This range is chosen to satisfy the conditions from ([22], Equation (29)) for condition [A].

Figures 1 and 2a–c show the simulation results for \mathcal{B} , measured θ_B , \mathcal{W} , and \mathcal{D} . θ_B was measured as the first angle in which $\mathcal{B}(\mathbf{h}, \mathbf{d}_{\theta_B/2}) \leq 0.05$. All methods manage to achieve the nulls in the desired directions from the LCMV. Observing Figure 2a, all but [E] are capable of maintaining a reasonable beamwidth across all frequencies, with [E] not being able to maintain the beamwidth because of too few sensors in the CB beamformer. The "naïve" combination of KP and LCKP [E, F] leads to worse performance for all metrics; thus, their results are not further compared.

Both methods (the RCKP and LCKP) lead to very akin results for θ_B , even though the LCKP simulations [C, D] have many more sensors in the CB beamformer than the RCKP ones [A, B]. This indicates that increasing the CB array size does not result in a better performance in either FNBW or directivity. However, in Figure 1, it can be seen that it generates a more focused beam. Both the RCKP and LCKP have similar WNG results of around 10 dB. The proposed method (especially [A]) leads to a better performance in terms of DF for all frequencies compared to the LCKP. The LCKP is marginally better for WNG in lower frequencies and does not suffer from performance loss at approximately 6.4kHz, caused by using rectangular arrays.

Comparing [A] and [B], the former has better results for both WNG and DF, while both have a similar result in maintaining a constant θ_B . This is a direct result of using the SD and DS beamformers in [A], and although [A] has less than half the number of sensors in the CB beamformer than [B], the FNBWs are similar. The beampattern of [B] is more focused than that of [A], but this does not cause a better performance in either beamwidth or directivity.

Comparing [C] and [D], both lead to almost identical results for all metrics. This can also be seen by comparing Figure 1c,d, where their beampatterns are almost indistinguishable. This indicates that the SD and DS beamformers are obfuscated by the CB in [C, D], caused by the CB beamformer having too many sensors compared to the other beamformers.

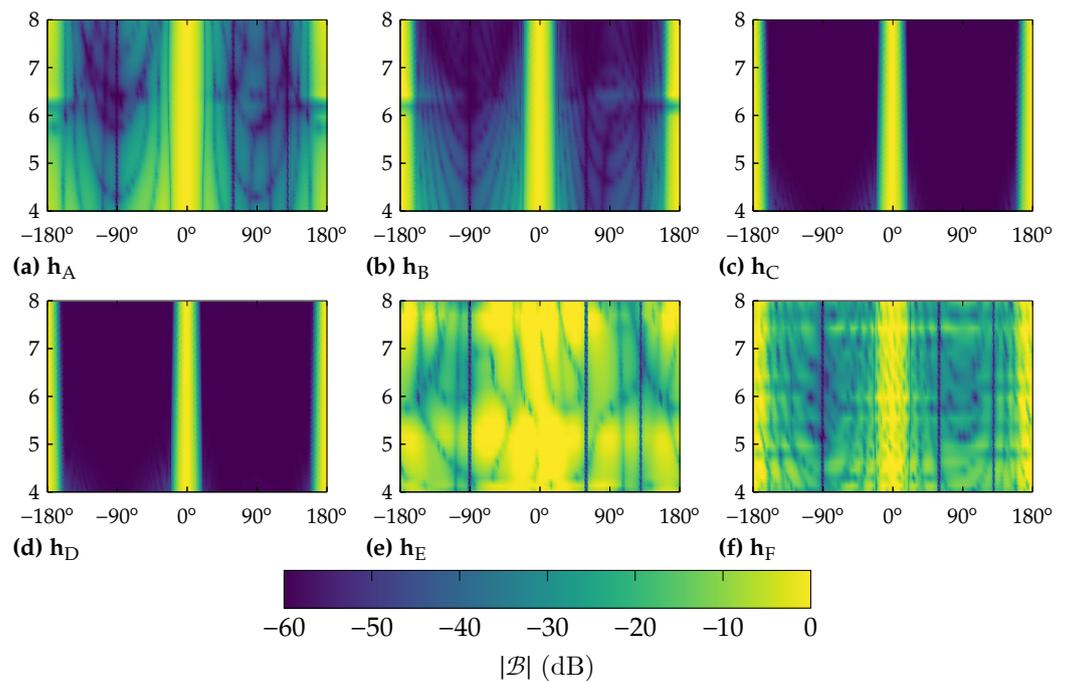


Figure 1. Beampattern heatmap for all situations in Table 1. x-axis is direction of source (in °), and y-axis is the frequency (in kHz).

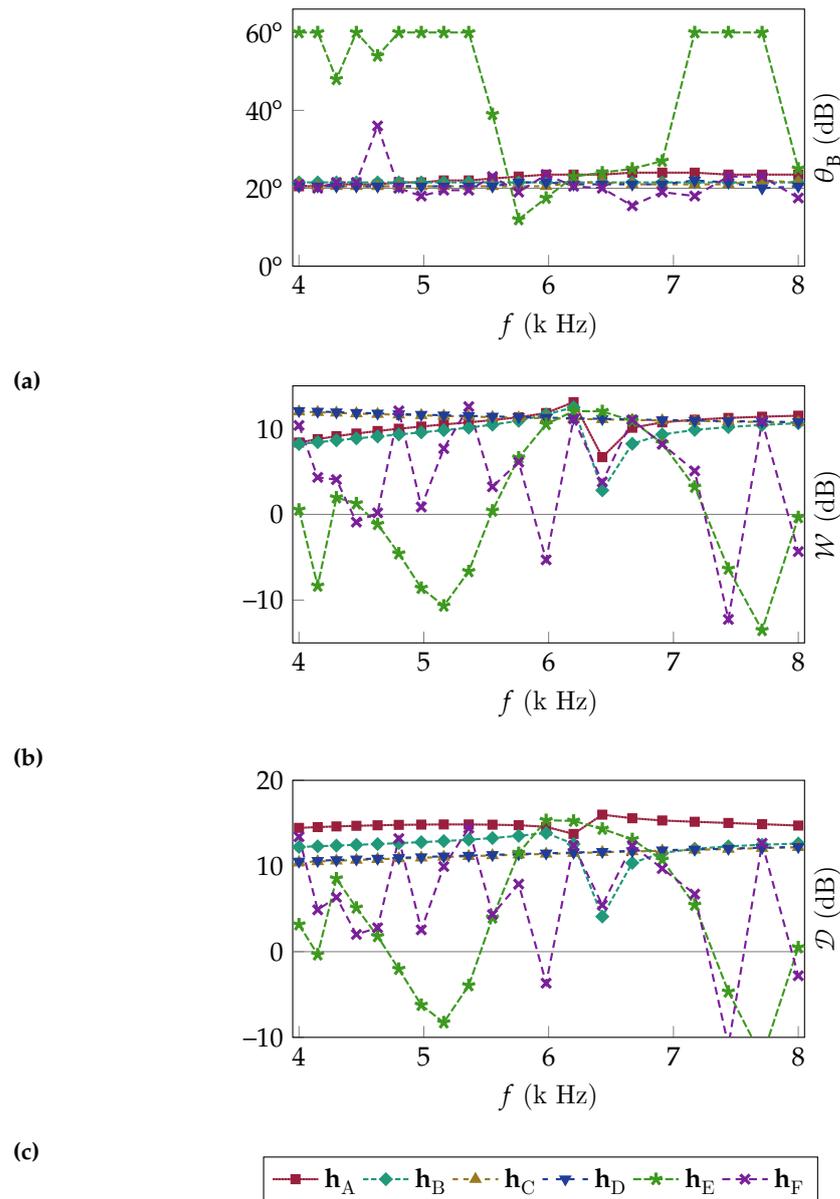


Figure 2. (a) FNBW, (b) WNG, and (c) DF for the beamformers designed with the parameters in Table 1.

6. Conclusions

We have introduced a novel approach for designing a constant-beamwidth beamformer with null-direction constraints, utilizing uniform rectangular arrays and a generalized convolutional Kronecker product beamforming technique. By synthesizing beamformers for virtual arrays and applying our proposed technique to a full array, we successfully achieved the desired features in terms of beamwidth and null placement. Moreover, by using virtual arrays and beamformers, we demonstrated the ability to enhance the signal quality in terms of white noise gain and the directivity factor without compromising the beamwidth. The experimental results using simulated sensor arrays demonstrate that the performance of our method surpassed the performance obtained using only known techniques based on the Kronecker product for beamforming synthesis when assessing beamwidth, white noise gain, and directivity.

Author Contributions: Conceptualization, I.C. and V.P.C.; Methodology, V.P.C.; Software, V.P.C.; Writing—original draft: V.P.C.; Writing—review and editing, I.C. and V.P.C.; Supervision, V.P.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Pazy Research Foundation and the Israel Science Foundation (grant no. 1449/23).

Data Availability Statement: The source code for the simulations developed herein is available at <https://github.com/VCurtarelli/py-cb-lcmv-rect> (accessed on 27 July 2023).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CB	Constant beamwidth
CKP	Convolutional KP
DF	Directivity factor
DS	Delay and sum
DSDI	Desired signal distortion index
FNBW	First-null beamwidth
KP	Kronecker product
LCKP	Linear CKP
LCMV	Linearly constrained minimum variance
MVDR	Minimum variance distortionless response
RCKP	Rectangular CKP
SD	Superdirective
ULA	Uniform linear array
URA	Uniform rectangular array
WNG	White noise gain

Appendix A. Proof of CKP Beamforming for URAs

We assume a rectangular uniform sensor array of size $M_x \times M_y$, with steering vector \mathbf{d} , and two subarrays of sizes $M_{1,x} \times M_{1,y}$ and $M_{2,x} \times M_{2,y}$, with steering vectors \mathbf{d}_1 and \mathbf{d}_2 , such that $M_{1,x} + M_{2,x} - 1 = M_x$ and $M_{1,y} + M_{2,y} - 1 = M_y$. We define

$$\tilde{\mathbf{h}} = \mathbf{h}_1 \otimes \mathbf{h}_2, \tag{A1a}$$

$$\tilde{\mathbf{d}} = \mathbf{d}_1 \otimes \mathbf{d}_2. \tag{A1b}$$

From this, it easily follows that

$$\tilde{\mathbf{h}}^H \tilde{\mathbf{d}} = (\mathbf{h}_1^H \mathbf{d}_1) (\mathbf{h}_2^H \mathbf{d}_2). \tag{A2}$$

Expanding both,

$$\mathbf{h}^H \mathbf{d} = \left(\sum_{m_1=0}^{M_1-1} h_1^*[m_1] d[m_1] \right) \left(\sum_{m_2=0}^{M_2-1} h_2^*[m_2] d[m_2] \right). \tag{A3}$$

We define $\mathbf{H}_1 = \mathcal{V}^{-1}(\mathbf{h}_1; M_{1,y})$ as the inverse vectorization of \mathbf{h}_1 ; similarly, $\mathbf{D} = \mathcal{V}^{-1}(\mathbf{d}; M_y)$ is the inverse vectorization of \mathbf{d} , and $\mathbf{H}_2 = \mathcal{V}^{-1}(\mathbf{h}_2; M_{2,y})$ is the inverse vectorization of \mathbf{h}_2 . Using the inverse vectorization on the sum over m_1 , we have

$$\sum_{m_1=0}^{M_1-1} h_1^*[m_1] d[m_1] = \sum_{m_{1,x}=0}^{M_{1,x}-1} \sum_{m_{1,y}=0}^{M_{1,y}-1} H_1^*[m_{1,x}, m_{1,y}] D[m_{1,x}, m_{1,y}]. \tag{A4}$$

We also assume that $D[m_{1,x}, m_{1,y}] = d_x[m_{1,x}]d_y[m_{1,y}]$, by the definition of the steering vector. Applying a similar process to the sum over m_2 from (A3),

$$\begin{aligned} \tilde{\mathbf{h}}^H \tilde{\mathbf{d}} &= \left(\sum_{\substack{m_{1,y}=0 \\ m_{1,x}=0}}^{M_{1,y}-1} H_1^*[m_{1,x}, m_{1,y}] d_x[m_{1,x}] d_y[m_{1,y}] \right) \\ &\times \left(\sum_{\substack{m_{2,y}=0 \\ m_{2,x}=0}}^{M_{2,y}-1} H_1^*[m_{2,x}, m_{2,y}] d_x[m_{2,x}] d_y[m_{2,y}] \right), \end{aligned} \tag{A5}$$

By applying the Cauchy product [33] to the second sum twice,

$$\begin{aligned} \tilde{\mathbf{h}}^H \tilde{\mathbf{d}} &= \sum_{\substack{m_y=0 \\ m_x=0}}^{M_{1,y}+M_{2,y}-2} \sum_{\substack{n_y=k_{1,y} \\ n_x=k_{1,x}}}^{k_{2,y}} (H_1^*[n_x, n_y] d_x[n_x] d_y[n_y]) \\ &\times (H_2^*[m_x - n_x, m_y - n_y] d_x[m_x - n_x] d_y[m_y - n_y]) \\ &= \sum_{\substack{m_y=0 \\ m_x=0}}^{M_x-1} \sum_{\substack{n_y=k_{1,y} \\ n_x=k_{1,x}}}^{k_{2,y}} (H_1^*[n_x, n_y] H_2^*[m_x - n_x, m_y - n_y]) \\ &\times (d_x[n_x] d_y[n_y] d_x[m_x - n_x] d_y[m_y - n_y]). \end{aligned} \tag{A6}$$

By the definition of the steering vectors, $d_x[a]d_x[b] = d_x[a + b]$ (also for d_y), and, therefore, the d term becomes $d_x[m_x]d_y[m_y] = D[m_x, m_y]$. By noting that the sum over \mathbf{H}_1 and \mathbf{H}_2 is the 2-D convolution between them (at the element $[m_x, m_y]$), and with this defining $\mathbf{H} = \mathbf{H}_1 \circledast \mathbf{H}_2$ (where \circledast denotes the 2-D convolution), then

$$\tilde{\mathbf{h}}^H \tilde{\mathbf{d}} = \sum_{\substack{m_y=0 \\ m_x=0}}^{M_x-1} H^*[m_x, m_y] D[m_x, m_y]. \tag{A7}$$

Finally, by vectorizing \mathbf{H} and \mathbf{D} into \mathbf{h} and \mathbf{d} ,

$$\begin{aligned} \tilde{\mathbf{h}}^H \tilde{\mathbf{d}} &= \sum_{\substack{m_y=0 \\ m_x=0}}^{M_x-1} h^*[M_y m_x + m_y] d[M_y m_x + m_y] \\ &= \sum_{m=0}^{M-1} h^*[m] d[m] \\ &= \mathbf{h}^H \mathbf{d}, \end{aligned} \tag{A8}$$

which concludes the proof.

Appendix B. Pseudocode Algorithms

These algorithms are written in a Python-like pseudolanguage. Brackets can denote both vector definition and vector slicing/indexing. For example, $a = [1, 2, 3, 4, 5]$ denotes a 1×5 vector, while $b = a[0 : 3]$ denotes a 1×3 vector, such that $b = [1, 2, 3]$ (last exclusive slicing, zero indexing). Comments are added where necessary to clarify the steps.

Algorithm A1 RCKP beamforming algorithm

Input:
 $\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_K$ # Input beamformers
 $M_{1,y}, M_{2,y}, \dots, M_{K,y}$ # Beamformer y-axis sizes

Output:
 \mathbf{h} # Full-array beamformer
 M_x, M_y # Output beamformer sizes

Procedure:
 $\mathbf{H} \leftarrow [[1]]$ # 1×1 matrix
 $M_y \leftarrow 1$
for $0 \leq k < K$ **do**
 $\mathbf{H}' \leftarrow \mathcal{V}^{-1}(\mathbf{h}_k; M_{k,y})$
 $\mathbf{H} \leftarrow \mathbf{H} \otimes \mathbf{H}'$
 $M_y \leftarrow M_y + M_{k,y} - 1$
end for
 $\mathbf{h} \leftarrow \mathcal{V}(\mathbf{H})$
 $M \leftarrow \text{len}(\mathbf{h})$ # Length of output vector
 $M_x \leftarrow \frac{M}{M_y}$

Algorithm A2 CB-LCMV beamformer algorithm

Input:
 $M_{1,x}, M'_{2,x}, M''_{2,x}, M_{1,y}, M_{2,y}$ # Array sizes
 $\mathbf{d}_{\theta_1;\text{LCMV}}, \dots, \mathbf{d}_{\theta_N;\text{LCMV}}, N$ # LCMV nulls
 $\mathbf{d}_x, \mathbf{d}_y$ # Steering vectors
 Φ_y, \mathbf{G} # Coherence matrices
 β # CB parameter

Output:
 \mathbf{h} # Full-array beamformer

Procedure:
LCMV beamformer
 $\mathbf{d}_{\text{LCMV}} \leftarrow \mathbf{d}_x[0 : M_{1,x}] \otimes \mathbf{d}_y[0 : M_{1,y}]$
 $\mathbf{C} \leftarrow [\mathbf{d}_{\text{LCMV}} \mathbf{d}_{\theta_1;\text{LCMV}} \dots \mathbf{d}_{\theta_N;\text{LCMV}}]$
 $\mathbf{q} \leftarrow [1] + [0] * N; \mathbf{q} \leftarrow \mathbf{q}^T$
 $\mathbf{h}_{\text{LCMV}} \leftarrow \Phi_y^{-1} \mathbf{C} (\mathbf{C}^H \Phi_y^{-1} \mathbf{C})^{-1} \mathbf{q}$ # Equation (12a)

CB beamformer
 $\mathbf{d}_{\text{CB}} \leftarrow \mathbf{d}_y[0 : M_{2,y}]$
 $\mathbf{h}_{\text{CB}} \leftarrow [0] * M_{2,y}; \mathbf{h}_{\text{CB}} \leftarrow \mathbf{h}_{\text{CB}}^T$
for $0 \leq m < M_{2,y}$ **do**
 $\mathbf{h}_{\text{CB}}[m] \leftarrow \text{calcCB}(m, M_{2,x}, \beta)$ # calcCB() is as in Equation (13)
end for
 $\mathbf{h}_{\text{CB}} \leftarrow \mathbf{h}_{\text{CB}} / \sum \mathbf{h}_{\text{CB}}$

SD and DS beamformers
 $\mathbf{d}_{\text{SD}} \leftarrow \mathbf{d}_x[0 : M'_{2,x}]$
 $\mathbf{d}_{\text{DS}} \leftarrow \mathbf{d}_x[0 : M''_{2,x}]$
 $\mathbf{h}_{\text{SD}} \leftarrow \frac{\mathbf{G}^{-1} \mathbf{d}}{\mathbf{d}^H \mathbf{G}^{-1} \mathbf{d}}$ # Equation (15a)
 $\mathbf{h}_{\text{DS}} \leftarrow \frac{\mathbf{d}}{M''_{2,x}}$ # Equation (15b)

Full-array beamformer
 $\mathbf{h} \leftarrow \text{RCKP}(\mathbf{h}_{\text{LCMV}}, \mathbf{h}_{\text{SD}}, \mathbf{h}_{\text{DS}}, \mathbf{h}_{\text{CB}}; M_{1,y}, 1, 1, M_{2,y})$ # Equation (A1)

References

1. Viswanath, P.; Tse, D.; Laroia, R. Opportunistic beamforming using dumb antennas. *IEEE Trans. Inf. Theory* **2002**, *48*, 1277–1294. [\[CrossRef\]](#)
2. Herbordt, W.; Nakamura, S.; Kellermann, W. Joint Optimization of LCMV Beamforming and Acoustic Echo Cancellation for Automatic Speech Recognition. In Proceedings of the (ICASSP '05), IEEE International Conference on Acoustics, Speech, and Signal Processing, Philadelphia, PA, USA, 23–23 March 2005; Volume 3, pp. 77–80. [\[CrossRef\]](#)
3. Chiariotti, P.; Martarelli, M.; Castellini, P. Acoustic beamforming for noise source localization—Reviews, methodology and applications. *Mech. Syst. Signal Process.* **2019**, *120*, 422–448. [\[CrossRef\]](#)
4. Haykin, S.; Liu, K.J.R. *Handbook on Array Processing and Sensor Networks*; Wiley-IEEE Press: New York, NY, USA, 2009.
5. Van Veen, B.; Buckley, K. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Mag.* **1988**, *5*, 4–24. [\[CrossRef\]](#)
6. Liu, W.; Weiss, S. *Wideband Beamforming Concepts and Techniques*; Wiley: New York, NY, USA, 2010.
7. Huang, Q.; Lin, M.; Wang, J.B.; Tsiftsis, T.A.; Wang, J. Energy Efficient Beamforming Schemes for Satellite-Aerial-Terrestrial Networks. *IEEE Trans. Commun.* **2020**, *68*, 3863–3875. [\[CrossRef\]](#)
8. Elbir, A.M.; Mishra, K.V.; Vorobyov, S.A.; Heath, R.W. Twenty-Five Years of Advances in Beamforming: From convex and nonconvex optimization to learning techniques. *IEEE Signal Process. Mag.* **2023**, *40*, 118–131. [\[CrossRef\]](#)
9. Gu, P.; Wang, G.; Fan, Z.; Chen, R. An Efficient Approach for the Synthesis of Large Sparse Planar Array. *IEEE Trans. Antennas Propag.* **2019**, *67*, 7320–7330. [\[CrossRef\]](#)
10. Zhang, X.; Zheng, W.; Chen, W.; Shi, Z. Two-dimensional DOA estimation for generalized coprime planar arrays: A fast-convergence trilinear decomposition approach. *Multidimens. Syst. Signal Process.* **2019**, *30*, 239–256. [\[CrossRef\]](#)
11. Lin, Z.; Lin, M.; Champagne, B.; Zhu, W.P.; Al-Dhahir, N. Secrecy-Energy Efficient Hybrid Beamforming for Satellite-Terrestrial Integrated Networks. *IEEE Trans. Commun.* **2021**, *69*, 6345–6360. [\[CrossRef\]](#)
12. Heidenreich, P.; Zoubir, A.M.; Rubsamen, M. Joint 2-D DOA Estimation and Phase Calibration for Uniform Rectangular Arrays. *IEEE Trans. Signal Process.* **2012**, *60*, 4683–4693. [\[CrossRef\]](#)
13. Ioannides, P.; Balanis, C. Uniform circular and rectangular arrays for adaptive beamforming applications. *IEEE Antennas Wirel. Propag. Lett.* **2005**, *4*, 351–354. [\[CrossRef\]](#)
14. Singh, S. Minimal Redundancy Linear Array and Uniform Linear Arrays Beamforming Applications in 5G Smart Devices. *Emerg. Sci. J.* **2021**, *4*, 70–84. [\[CrossRef\]](#)
15. Goodwin, M.; Elko, G. Constant beamwidth beamforming. In Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing, Minneapolis, MN, USA, 27–30 April 1993; Volume 1, pp. 169–172. [\[CrossRef\]](#)
16. Zarifi, K.; Affes, S.; Ghayeb, A. Collaborative Null-Steering Beamforming for Uniformly Distributed Wireless Sensor Networks. *IEEE Trans. Signal Process.* **2010**, *58*, 1889–1903. [\[CrossRef\]](#)
17. Frost, O. An algorithm for linearly constrained adaptive array processing. *Proc. IEEE* **1972**, *60*, 926–935. [\[CrossRef\]](#)
18. Buckley, K. Spatial/Spectral filtering with linearly constrained minimum variance beamformers. *IEEE Trans. Acoust. Speech Signal Process.* **1987**, *35*, 249–266. [\[CrossRef\]](#)
19. Souden, M.; Benesty, J.; Affes, S. A Study of the LCMV and MVDR Noise Reduction Filters. *IEEE Trans. Signal Process.* **2010**, *58*, 4925–4935. [\[CrossRef\]](#)
20. Hixson, E.L.; Au, K.T. Wide-Bandwidth Constant-Beamwidth Acoustic Array. *J. Acoust. Soc. Am.* **1970**, *48*, 117–117. [\[CrossRef\]](#)
21. Wang, Z.; Li, J.; Stoica, P.; Nishida, T.; Sheplak, M. Constant-beamwidth and constant-powerwidth wideband robust Capon beamformers for acoustic imaging. *J. Acoust. Soc. Am.* **2004**, *116*, 1621–1631. [\[CrossRef\]](#)
22. Long, T.; Cohen, I.; Berdugo, B.; Yang, Y.; Chen, J. Window-Based Constant Beamwidth Beamformer. *Sensors* **2019**, *19*, 2091. [\[CrossRef\]](#)
23. Frank, A.; Ben-Kish, A.; Cohen, I. Constant-Beamwidth Linearly Constrained Minimum Variance Beamformer. In Proceedings of the 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 29 August–2 September 2022; pp. 50–54. [\[CrossRef\]](#)
24. Benesty, J.; Chen, J.; Huang, Y. *Microphone Array Signal Processing*; Springer Topics in Signal Processing; Springer: Berlin/Heidelberg, Germany, 2008; Volume 1. [\[CrossRef\]](#)
25. Abramovich, Y.I.; Frazer, G.J.; Johnson, B.A. Iterative Adaptive Kronecker MIMO Radar Beamformer: Description and Convergence Analysis. *IEEE Trans. Signal Process.* **2010**, *58*, 3681–3691. [\[CrossRef\]](#)
26. Werner, K.; Jansson, M.; Stoica, P. On Estimation of Covariance Matrices With Kronecker Product Structure. *IEEE Trans. Signal Process.* **2008**, *56*, 478–491. [\[CrossRef\]](#)
27. Habets, E.A.P.; Gannot, S. Generating sensor signals in isotropic noise fields. *J. Acoust. Soc. Am.* **2007**, *122*, 3464–3470. [\[CrossRef\]](#)
28. Markovich-Golan, S.; Gannot, S.; Kellermann, W. Combined LCMV-TRINICON Beamforming for Separating Multiple Speech Sources in Noisy and Reverberant Environments. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2017**, *25*, 320–332. [\[CrossRef\]](#)
29. Erdogan, H.; Hershey, J.R.; Watanabe, S.; Mandel, M.I.; Roux, J.L. Improved MVDR Beamforming Using Single-Channel Mask Prediction Networks. In Proceedings of the Interspeech 2016, ISCA—17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, 8–12 September 2016; pp. 1981–1985. [\[CrossRef\]](#)
30. Kaiser, J.; Schafer, R. On the use of the 10-sinh window for spectrum analysis. *IEEE Trans. Acoust. Speech Signal Process.* **1980**, *28*, 105–107. [\[CrossRef\]](#)

31. Brandstein, M.; Ward, D.; Lacroix, A.; Venetsanopoulos, A. (Eds.) *Microphone Arrays: Signal Processing Techniques and Applications*; Digital Signal Processing; Springer: Berlin/Heidelberg, Germany, 2001. [[CrossRef](#)]
32. Huang, G.; Benesty, J.; Chen, J.; Cohen, I. Robust and steerable kronecker product differential beamforming With rectangular microphone arrays. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 211–215. [[CrossRef](#)]
33. Hamahata, Y. Arithmetic functions and the Cauchy product. *Arch. Math.* **2020**, *114*, 41–50. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.