Article

Constant-Beamwidth LCMV Beamformer with Rectangular Arrays

Vitor Curtarelli[1,*], Israel Cohen[2]

[1] Graduate Program in Electrical Engineering, Universidade Federal de Santa Catarina, Florianópolis, SC, Brasil; vitor.curtarelli@gmail.com
[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 the physical array into virtual subarrays, each tailored to achieve a specific desired feature, and 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. Comparative analysis against existing methods from the literature reveals that the proposed method performs better than 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 since, in this case, frequency does not affect the behavior of the beamformer. Some desired features are constant-beamwidth [15] and null steering [16]. One approach for null steering is using the 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. On the other hand, constant-beamwidth (CB) beamforming [15,20,21] can be accomplished by using window-based beamforming techniques [22], but this method cannot incorporate directional restrictions. CB-LCMV

Citation: Curtarelli, V.; Cohen, I. Constant-Beamwidth LCMV Beamformer with Rectangular Arrays. *Algorithms* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

beamformers have been explored recently [23], however only in the context of linear sensor arrays, leaving space for their exploration in the context of different array configurations.

The delay-and-sum (DS) and superdirective (SD) beamformers [24] can be used to increase white noise gain or directivity factor, respectively, by maximizing the desired metric. 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 the 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 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 (KP and LCKP), beampattern features and distortionless constraints are conserved in the combined beamformers. Therefore, by using beamformers with desired beampattern 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. For such, 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 directivity factor. The performance of the proposed method is compared against beamformers obtained through the KP and LCKP methods. Our results demonstrate superior performance in terms of beamwidth, white noise gain, and directivity for the beamformers obtained using the proposed method when compared to 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 will be used further. In Section 4, the newly proposed method for 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 simulations realized and discuss the results, comparing them to the literature. Finally, Section 6 concludes this paper, overviewing the main contributions.

2. Signal and Array Model

Let S be a uniform rectangular array (URA) of sensors over the $\mathrm{x}-\mathrm{y}$ 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 $M = M_{\mathrm{x}}M_{\mathrm{y}}$ sensors. Assume a source in the far-field on the same plane as the sensor array (that is, with elevation $\phi = 0^{\mathrm{o}}$), 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^{o}. This constraint can be easily removed without affecting the mathematical framework developed.

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

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

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 $(m_x\delta_\mathrm{x}, m_y\delta_\mathrm{y})$, $\omega = 2\pi f$ is the angular frequency, f is the temporal frequency, and $\mathrm{j} = \sqrt{-1}$ is the imaginary unit. We denote by $\mathbf{d}(\omega, \theta) = \mathcal{V}(\mathbf{D}(\omega, \theta))$ the $M \times 1$ steering vector, with $M = M_x M_y$, and $\mathcal{V}(\,\cdot\,)$ being the vectorization operation; and let $\mathbf{D}(\omega, \theta) = \mathcal{V}^{-1}(\mathbf{d}(\omega, \theta);\, M_\mathrm{y})$ denote the inverse vectorization of $\mathbf{d}(\omega, \theta)$.

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

$$\mathbf{y}(\omega) = \mathbf{d}(\omega, \theta_\mathrm{d}) X(\omega) + \mathbf{v}(\omega), \tag{2}$$

where $X(\omega)$ is the desired signal at the reference sensor, $\mathbf{d}(\omega, \theta_\mathrm{d})$ is the steering vector of the desired source from the 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 desires signal $X(\omega)$ as $Z(\omega)$ using a beamformer $\mathbf{h}(\omega)$ (assumed a 2-D beamformer), through the linear filtering

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

where the superscript $^\mathsf{H}$ denotes the conjugate-transpose operator. The beamformer is called distortionless if it satisfies $\mathbf{h}(\omega)^\mathsf{H} \mathbf{d}(\omega, \theta_\mathrm{d}) = 1$ for all ω. In that case,

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

where $\mathbf{v}_\mathrm{rn}(\omega) = \mathbf{h}(\omega)^\mathsf{H} \mathbf{v}(\omega)$ is the residual noise at the beamformer's output. This constraint guarantees the beamformer to not affect the desired signal, only altering the undesired noise signal.

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

2.1. Beamformer metrics

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

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

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

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

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

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

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

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

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

where ν_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 $\mathsf{E}\left[\mathbf{u}^{\mathsf{H}}(\mathbf{d}_{\theta_n}\nu_n)\right] = 0$. 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

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

$$\mathbf{q} = [\, 1,\ 0,\ \cdots,\ 0\,]^{\mathsf{T}}, \tag{8b}$$

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

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

$$\mathbf{h}_{\mathrm{LCMV}} = \underset{\mathbf{h}}{\operatorname{argmin}}\, \mathbf{h}^{\mathsf{H}}\boldsymbol{\Phi}_{\mathbf{v}}\mathbf{h} \ \text{ s.t. } \ \mathbf{C}^{\mathsf{H}}\mathbf{h} = \mathbf{q}, \tag{9}$$

where $\boldsymbol{\Phi}_{\mathbf{v}}(\omega) = \mathsf{E}\left[\mathbf{v}(\omega)\mathbf{v}^{\mathsf{H}}(\omega)\right]$ 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 to the N undesired directions. The solution to this minimization problem is

$$\mathbf{h}_{\mathrm{LCMV}} = \boldsymbol{\Phi}_{\mathbf{v}}^{-1}\mathbf{C}\left[\mathbf{C}^{\mathsf{H}}\boldsymbol{\Phi}_{\mathbf{v}}^{-1}\mathbf{C}\right]^{-1}\mathbf{q}. \tag{10}$$

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 the minimum variance distortionless response (MVDR) beamformer [29], which is given by

$$\mathbf{h}_{\mathrm{MVDR}} = \frac{\boldsymbol{\Phi}_{\mathbf{v}}^{-1}\mathbf{d}}{\mathbf{d}^{\mathsf{H}}\boldsymbol{\Phi}_{\mathbf{v}}^{-1}\mathbf{d}}. \tag{11}$$

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

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

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

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 $|\theta - \theta_{\text{d}}| < \theta_{\text{B}}/2$. That is, $\theta_{\text{B}}/2$ 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 θ_{B} constant. This technique requires that the desired source signal is impinging 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

The 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

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

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

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 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 (or full-array) beamformer.

The KP beamforming process is as follows: given a steering vector $\mathbf{d}_\theta$, 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 $\mathbf{d}_\theta$ of length M, we define $\mathbf{d}_{1;\theta}$ as the M_1-th first elements of $\mathbf{d}_\theta$, 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:

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

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

The first one shows that the beampattern of $\mathbf{h}$ is the combination of the beampatterns for the subarrays. Through the second one, we can see that if $\mathbf{h}_1$ and $\mathbf{h}_2$ are distortionless beamformers, so will $\mathbf{h}$ 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 both the lack of symmetry from the rectangular arrays but 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 $M_{1,\mathrm{x}} \times M_{1,\mathrm{y}}$ first sensors of S, with a steering vector $\mathbf{d}_1$; and similarly for S_2. These arrays sizes are such that $M_{1,\mathrm{x}} + M_{2,\mathrm{x}} - 1 = M_{\mathrm{x}}$, and $M_{1,\mathrm{y}} + M_{2,\mathrm{y}} - 1 = M_{\mathrm{y}}$. 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 the beamformer for the full-array S through

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

with $\circledast$ 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 1 in Appendix B, which is written in a Python-like pseudolanguage.

In the same way as with KP and LCKP methods, with RCKP, one can design beamformers for the subarrays S_1 and S_2, and 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 we have a total of $M' = M_{1,\mathrm{x}} M_{1,\mathrm{y}} M_{2,\mathrm{x}} M_{2,\mathrm{y}}$ virtual sensors. Using that all Ms are ≥ 1, it is trivial to see that[1] $M' \geq M$. Therefore, we virtually have more sensors 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 loss of generalization. In this case, assuming we are using K beamformers, each with size $M_{k,\mathrm{x}} \times M_{k,\mathrm{y}}$, then their dimensions must be such that

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

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

This is easily verifiable by repeating the synthesis operation from Eq. (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 such, the full-array S is separated into four subarrays: S_1, $S'_{2;\mathrm{x}}$, $S''_{2;\mathrm{x}}$ and $S_{2;\mathrm{y}}$. Each subarray is used to design one of the desired beamformers presented in Sections 3.1 to 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 $M_{1,\mathrm{x}} \times M_{1,\mathrm{y}}$. As explained previously, this choice has the advantage of the rectangular array's lesser symmetry compared to a linear array. We have $M_1 = M_{1,\mathrm{x}} M_{1,\mathrm{y}}$ 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 $S_{2;\mathrm{y}}$ 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 $\theta_{\mathrm{d}} = 0^{\mathrm{o}}$.

The SD and DS beamformers are built from the $S'_{2;\mathrm{x}}$ and $S''_{2;\mathrm{x}}$ subarrays respectively, based on Section 3.3. We assume 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 the 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

$$\mathbf{h} = \mathcal{V}\Big(\mathcal{V}^{-1}\big(\mathbf{h}_{\mathrm{LCMV}};\, M_{1,\mathrm{y}}\big) \circledast \mathcal{V}^{-1}\big(\mathbf{h}_{\mathrm{SD}};\, 1\big) \circledast \mathcal{V}^{-1}\big(\mathbf{h}_{\mathrm{DS}};\, 1\big) \circledast \mathcal{V}^{-1}\big(\mathbf{h}_{\mathrm{CB}};\, M_{2,\mathrm{y}}\big)\Big). \tag{19}$$

[1] The equality happens if S_1 and S_2 are perpendicular ULAs, or one of them has only one sensor.

An implementation of the proposed CB-LCMV beamformer is shown in Algorithm 2 in Appendix B. The procedure calculates the 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 $|\theta - \theta_{\mathrm{d}}| < \theta_{\mathrm{B}} \,/\, 2$, where $\mathbf{h} \in [\mathbf{h}_{\mathrm{LCMV}}, \mathbf{h}_{\mathrm{CB}}, \mathbf{h}_{\mathrm{SD}}, \mathbf{h}_{\mathrm{DS}}]$. 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 FNBW is sufficient to satisfy the condition.

5. Experimental Results

In this section, we present simulations[2] performed to verify the proposed method implementation and compare its performance to those constructed from the existing methods. 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.

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

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

In all situations, S has a total of $M = 36$ sensors. The sensors are assumed to be ideal omnidirectional sensors with plain frequency response over the spectrum. The intersensor distances are $\delta_{\mathrm{x}} = 0.5$cm and $\delta_{\mathrm{y}} = 3.0$cm. We assume that $\theta_{\mathrm{d}} = 0^{\mathrm{o}}$ and $\theta_{\mathrm{B}} = 40^{\mathrm{o}}$. Since the LCMV has 4 sensors, we will use 3 interfering sources (for all situations), with directions $\theta_i \in [-90^{\mathrm{o}}, 60^{\mathrm{o}}, 130^{\mathrm{o}}]$, each with $\phi_{\mathbf{v}_i} = 1$, and also assume $\mathbf{v}'$ to be uncorrelated Gaussian white noise with unit variance (that is, $\mathbf{\Phi}_{\mathbf{v}'}$ is the identity matrix). The variance of the desired signal is $\phi_X = \mathsf{E}\left[|X|^2\right] = 5$. The simulations are performed for the range $f \in [4, 8]$kHz. This range was chosen to satisfy the conditions from [22, Eq. (29)] for condition [A].

Figures 1, 2a, 2b and 2c show the simulation results for $\mathcal{B}$, measured[3] θ_{B}, $\mathcal{W}$ and $\mathcal{D}$, respectively. All methods managed to achieve the nulls in the desired directions from the LCMV. Observing Fig. 2a, all but [E] were 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 (RCKP and LCKP) led to very akin results for θ_{B}, even though the LCKP simulations [C, D] had much more sensors in the CB beamformer than the RCKP

[2] The code used for these simulations is available at https://github.com/VCurtarelli/py-cb-lcmv-rect.
[3] θ_{B} was measured as the first angle in which $\mathcal{B}(\mathbf{h}, \mathbf{d}_{\theta_{\mathrm{B}}/2}) \leq 0.05$.

ones [A, B]. This indicates that increasing the CB array size does not result in a better performance in either FNBW or directivity. However, from Fig. 1, it is seen that it generates a more focused beam. Both the RCKP and LCKP had similar WNG results of around 10dB. 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 Figs. 1c and 1d, 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.

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 the 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 sig-

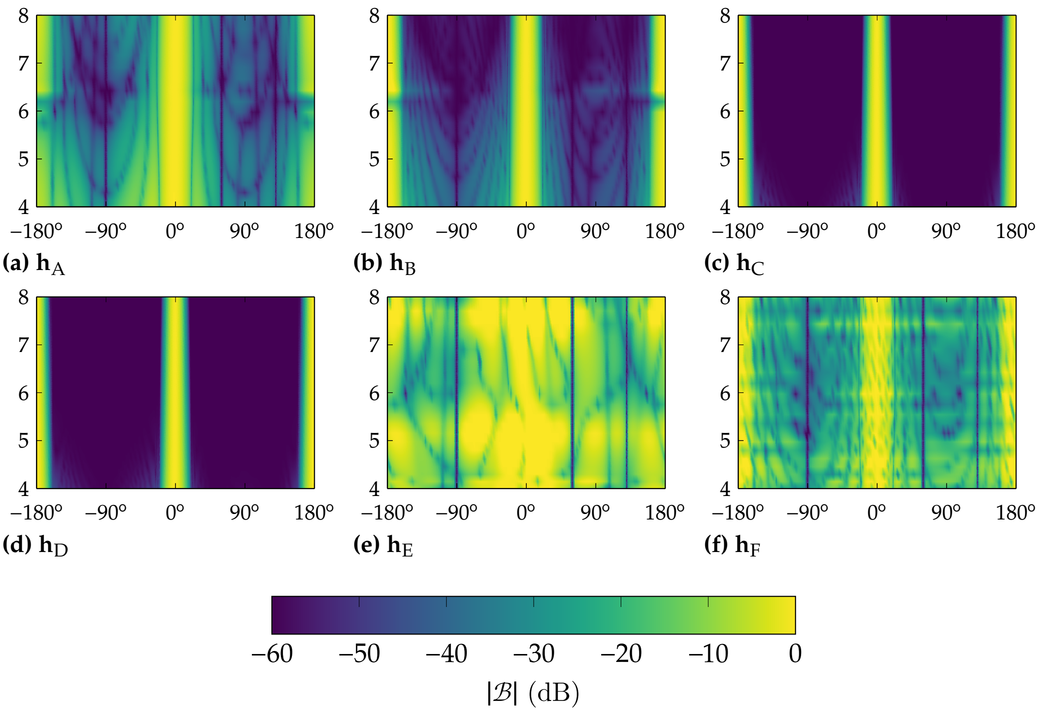

Figure 1. Beampattern heatmap for all situations in Table 1. x-axis is direction of source (in °), and y-axis the frequency (in kHz).

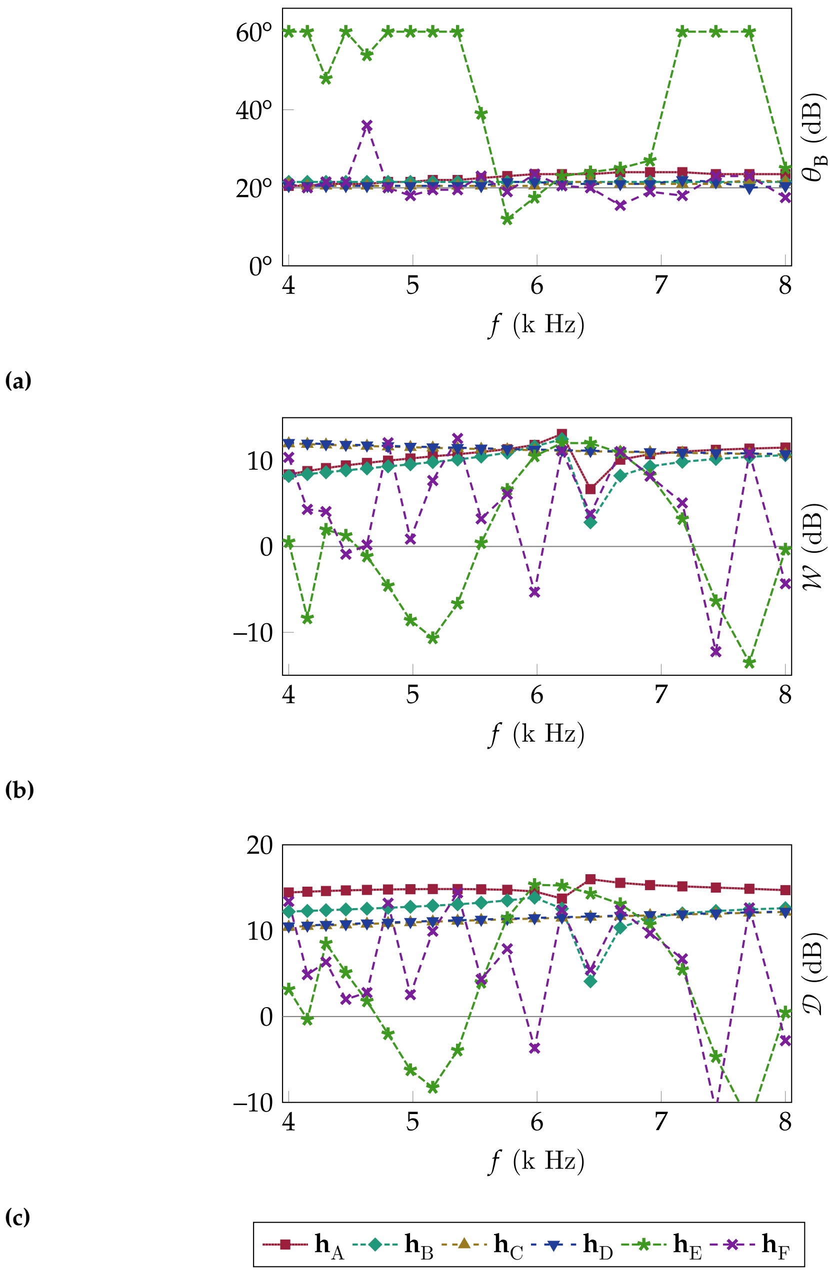

Figure 2. (a) FNBW, **(b)** WNG, and **(c)** DF, for the beamformers designed with with the parameters in Table 1.

nal quality in terms of white noise gain and directivity factor without compromising the beamwidth. Experimental results using simulated sensor arrays demonstrate that 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. Cohen and V. Curtarelli; Methodology, V. Curtarelli; Software, V. Curtarelli; Writing—original draft: V. Curtarelli; Writing—review and editing, I. Cohen and V. Curtarelli; Supervision, V. Curtarelli. 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 here is available at https://github.com/VCurtarelli/py-cb-lcmv-rect.

Conflicts of Interest: The authors declare no conflict 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 $M_{\mathrm{x}} \times M_{\mathrm{y}}$ with a steering vector $\mathbf{d}$, and two subarrays of sizes $M_{1,\mathrm{x}} \times M_{1,\mathrm{y}}$ and $M_{2,\mathrm{x}} \times M_{2,\mathrm{y}}$ with steering vectors $\mathbf{d}_1$ and $\mathbf{d}_2$, such that $M_{1,\mathrm{x}} + M_{2,\mathrm{x}} - 1 = M_{\mathrm{x}}$ and $M_{1,\mathrm{y}} + M_{2,\mathrm{y}} - 1 = M_{\mathrm{y}}$. We define

$$\widetilde{\mathbf{h}} = \mathbf{h}_1 \otimes \mathbf{h}_2, \tag{A1a}$$
$$\widetilde{\mathbf{d}} = \mathbf{d}_1 \otimes \mathbf{d}_2. \tag{A1b}$$

From this, it easily follows that

$$\widetilde{\mathbf{h}}^{\mathsf{H}} \widetilde{\mathbf{d}} = \left(\mathbf{h}_1^{\mathsf{H}} \mathbf{d}_1\right)\left(\mathbf{h}_2^{\mathsf{H}} \mathbf{d}_2\right). \tag{A2}$$

Expanding both,

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

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

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

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

$$\widetilde{\mathbf{h}}^{\mathsf{H}}\widetilde{\mathbf{d}} = \left(\sum_{m_{1,\mathrm{x}}=0}^{M_{1,\mathrm{x}}-1}\sum_{m_{1,\mathrm{y}}=0}^{M_{1,\mathrm{y}}-1} H_1^*[m_{1,\mathrm{x}}, m_{1,\mathrm{y}}]d_{\mathrm{x}}[m_{1,\mathrm{x}}]d_{\mathrm{y}}[m_{1,\mathrm{y}}]\right) \times \left(\sum_{m_{2,\mathrm{x}}=0}^{M_{2,\mathrm{x}}-1}\sum_{m_{2,\mathrm{y}}=0}^{M_{2,\mathrm{y}}-1} H_1^*[m_{2,\mathrm{x}}, m_{2,\mathrm{y}}]d_{\mathrm{x}}[m_{2,\mathrm{x}}]d_{\mathrm{y}}[m_{2,\mathrm{y}}]\right), \tag{A5}$$

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

$$\begin{aligned}\widetilde{\mathbf{h}}^{\mathsf{H}}\widetilde{\mathbf{d}} &= \sum_{m_{\mathrm{x}}=0}^{M_{1,\mathrm{x}}+M_{2,\mathrm{x}}-2}\sum_{m_{\mathrm{y}}=0}^{M_{1,\mathrm{y}}+M_{2,\mathrm{y}}-2}\ \sum_{n_{\mathrm{x}}=k_{1,\mathrm{x}}}^{k_{2,\mathrm{x}}}\sum_{n_{\mathrm{y}}=k_{1,\mathrm{y}}}^{k_{2,\mathrm{y}}} (H_1^*[n_{\mathrm{x}}, n_{\mathrm{y}}]d_{\mathrm{x}}[n_{\mathrm{x}}]d_{\mathrm{y}}[n_{\mathrm{y}}]) \\ &\quad \times (H_2^*[m_{\mathrm{x}}-n_{\mathrm{x}}, m_{\mathrm{y}}-n_{\mathrm{y}}]d_{\mathrm{x}}[m_{\mathrm{x}}-n_{\mathrm{x}}]d_{\mathrm{y}}[m_{\mathrm{y}}-n_{\mathrm{y}}]) \\ &= \sum_{m_{\mathrm{x}}=0}^{M_{\mathrm{x}}-1}\sum_{m_{\mathrm{y}}=0}^{M_{\mathrm{y}}-1}\ \sum_{n_{\mathrm{x}}=k_{1,\mathrm{x}}}^{k_{2,\mathrm{x}}}\sum_{n_{\mathrm{y}}=k_{1,\mathrm{y}}}^{k_{2,\mathrm{y}}} (H_1^*[n_{\mathrm{x}}, n_{\mathrm{y}}]H_2^*[m_{\mathrm{x}}-n_{\mathrm{x}}, m_{\mathrm{y}}-n_{\mathrm{y}}]) \\ &\quad \times (d_{\mathrm{x}}[n_{\mathrm{x}}]d_{\mathrm{y}}[n_{\mathrm{y}}]d_{\mathrm{x}}[m_{\mathrm{x}}-n_{\mathrm{x}}]d_{\mathrm{y}}[m_{\mathrm{y}}-n_{\mathrm{y}}]).\end{aligned} \tag{A6}$$

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

$$\widetilde{\mathbf{h}}^{\mathsf{H}}\widetilde{\mathbf{d}} = \sum_{m_{\mathrm{x}}=0}^{M_{\mathrm{x}}-1}\sum_{m_{\mathrm{y}}=0}^{M_{\mathrm{y}}-1} H^*[m_{\mathrm{x}}, m_{\mathrm{y}}]D[m_{\mathrm{x}}, m_{\mathrm{y}}]. \tag{A7}$$

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

$$\begin{aligned}\widetilde{\mathbf{h}}^{\mathsf{H}}\widetilde{\mathbf{d}} &= \sum_{m_{\mathrm{x}}=0}^{M_{\mathrm{x}}-1}\sum_{m_{\mathrm{y}}=0}^{M_{\mathrm{y}}-1} h^*[M_{\mathrm{y}}m_{\mathrm{x}}+m_{\mathrm{y}}]d[M_{\mathrm{y}}m_{\mathrm{x}}+m_{\mathrm{y}}] \\ &= \sum_{m=0}^{M-1} h^*[m]d[m] \\ &= \mathbf{h}^{\mathsf{H}}\mathbf{d},\end{aligned} \tag{A8}$$

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 1 RCKP beamforming algorithm

Input:
$\mathbf{h}_1 \,,\, \mathbf{h}_2 \,,\, \cdots \,,\, \mathbf{h}_K$ — *# Input beamformers*
$M_{1,\mathrm{y}} \,,\, M_{2,\mathrm{y}} \,,\, \cdots \,,\, M_{K,\mathrm{y}}$ — *# Beamformer y-axis sizes*
Output:
$\mathbf{h}$ — *# Full-array beamformer*
$M_\mathrm{x} \,,\, M_\mathrm{y}$ — *# Output beamformer sizes*
Procedure:
$\mathbf{H} \leftarrow [[1]]$ — *# 1×1 matrix*
$M_\mathrm{y} \leftarrow 1$
for $0 \leq k < K$ **do**
  $\mathbf{H}' \leftarrow \mathcal{V}^{-1}(\mathbf{h}_k;\, M_{k,\mathrm{y}})$
  $\mathbf{H} \leftarrow \mathbf{H} \circledast \mathbf{H}'$
  $M_\mathrm{y} \leftarrow M_\mathrm{y} + M_{k,\mathrm{y}} - 1$
end for
$\mathbf{h} \leftarrow \mathcal{V}(\mathbf{H})$
$M \leftarrow \mathrm{len}(\mathbf{h})$ — *# Length of output vector*
$M_\mathrm{x} \leftarrow \frac{M}{M_\mathrm{y}}$

Algorithm 2 CB-LCMV beamformer algorithm

Input:
$M_{1,\mathrm{x}} \,,\, M'_{2,\mathrm{x}} \,,\, M''_{2,\mathrm{x}} \,,\, M_{1,\mathrm{y}} \,,\, M_{2,\mathrm{y}}$ — *# Array sizes*
$\mathbf{d}_{\theta_1;\mathrm{LCMV}} \,,\, \cdots \,,\, \mathbf{d}_{\theta_N;\mathrm{LCMV}} \,,\, N$ — *# LCMV nulls*
$\mathbf{d}_\mathrm{x} \,,\, \mathbf{d}_\mathrm{y}$ — *# Steering vectors*
$\mathbf{\Phi}_\mathbf{y} \,,\, \mathbf{G}$ — *# Coherence matrices*
β — *# CB parameter*
Output:
$\mathbf{h}$ — *# Full-array beamformer*
Procedure:
 # LCMV beamformer
$\mathbf{d}_\mathrm{LCMV} \leftarrow \mathbf{d}_\mathrm{x}[0 : M_{1,\mathrm{x}}] \otimes \mathbf{d}_\mathrm{y}[0 : M_{1,\mathrm{y}}]$
$\mathbf{C} \leftarrow [\mathbf{d}_\mathrm{LCMV}\ \mathbf{d}_{\theta_1;\mathrm{LCMV}}\ \cdots\ \mathbf{d}_{\theta_N;\mathrm{LCMV}}]$
$\mathbf{q} \leftarrow [1] + [0] * N;\ \mathbf{q} \leftarrow \mathbf{q}^\mathrm{T}$
$\mathbf{h}_\mathrm{LCMV} \leftarrow \mathbf{\Phi}_\mathbf{y}^{-1}\mathbf{C}\left(\mathbf{C}^\mathrm{H}\mathbf{\Phi}_\mathbf{y}^{-1}\mathbf{C}\right)^{-1}\mathbf{q}$ — *# Eq. (12a)*
 # CB beamformer
$\mathbf{d}_\mathrm{CB} \leftarrow \mathbf{d}_\mathrm{y}[0 : M_{2,\mathrm{y}}]$
$\mathbf{h}_\mathrm{CB} \leftarrow [0] * M_{2,\mathrm{y}};\ \mathbf{h}_\mathrm{CB} \leftarrow \mathbf{h}_\mathrm{CB}^\mathrm{T}$
for $0 \leq m < M_{2,\mathrm{y}}$ **do**
  $\mathbf{h}_\mathrm{CB}[m] \leftarrow \mathrm{calcCB}(m, M_{2,\mathrm{x}}, \beta)$ — *# calcCB() is as in Eq. (13)*
end for
$\mathbf{h}_\mathrm{CB} \leftarrow \mathbf{h}_\mathrm{CB} / \sum \mathbf{h}_\mathrm{CB}$
 # SD and DS beamformers
$\mathbf{d}_\mathrm{SD} \leftarrow \mathbf{d}_\mathrm{x}[0 : M'_{2,\mathrm{x}}]$
$\mathbf{d}_\mathrm{DS} \leftarrow \mathbf{d}_\mathrm{x}[0 : M''_{2,\mathrm{x}}]$
$\mathbf{h}_\mathrm{SD} \leftarrow \frac{\mathbf{G}^{-1}\mathbf{d}}{\mathbf{d}^\mathrm{H}\mathbf{G}^{-1}\mathbf{d}}$ — *# Eq. (15a)*
$\mathbf{h}_\mathrm{DS} \leftarrow \frac{\mathbf{d}}{M''_{2,\mathrm{x}}}$ — *# Eq. (15b)*
 # Full-array beamformer
$\mathbf{h} \leftarrow \mathrm{RCKP}(\mathbf{h}_\mathrm{LCMV}, \mathbf{h}_\mathrm{SD}, \mathbf{h}_\mathrm{DS}, \mathbf{h}_\mathrm{CB}\ ;\ M_{1,\mathrm{y}}, 1, 1, M_{2,\mathrm{y}})$ — *# Algorithm 1*

References

1. Viswanath, P.; Tse, D.; Laroia, R. Opportunistic beamforming using dumb antennas. *IEEE Transactions on Information Theory* **2002**, *48*, 1277–1294. https://doi.org/10.1109/TIT.2002.1003822.
2. Herbordt, W.; Nakamura, S.; Kellermann, W. Joint Optimization of LCMV Beamforming and Acoustic Echo Cancellation for Automatic Speech Recognition. In Proceedings of the Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005., Philadelphia, Pennsylvania, USA, 2005; Vol. 3, pp. 77–80. https://doi.org/10.1109/ICASSP.2005.1415650.
3. Chiariotti, P.; Martarelli, M.; Castellini, P. Acoustic beamforming for noise source localization Reviews, methodology and applications. *Mechanical Systems and Signal Processing* **2019**, *120*, 422–448. https://doi.org/10.1016/j.ymssp.2018.09.019.
4. Haykin, S.; Liu, K.J.R. Handbook on Array Processing and Sensor Networks **2009**.
5. Van Veen, B.; Buckley, K. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine* **1988**, *5*, 4–24. https://doi.org/10.1109/53.665.
6. Liu, W.; Weiss, S. Wideband Beamforming Concepts and Techniques **2010**.
7. Huang, Q.; Lin, M.; Wang, J.B.; Tsiftsis, T.A.; Wang, J. Energy Efficient Beamforming Schemes for Satellite-Aerial-Terrestrial Networks. *IEEE Transactions on Communications* **2020**, *68*, 3863–3875. https://doi.org/10.1109/TCOMM.2020.2978044.
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 Processing Magazine* **2023**, *40*, 118–131. https://doi.org/10.1109/MSP.2023.3262366.
9. Gu, P.; Wang, G.; Fan, Z.; Chen, R. An Efficient Approach for the Synthesis of Large Sparse Planar Array. *IEEE Transactions on Antennas and Propagation* **2019**, *67*, 7320–7330. https://doi.org/10.1109/TAP.2019.2931959.
10. Zhang, X.; Zheng, W.; Chen, W.; Shi, Z. Two-dimensional DOA estimation for generalized coprime planar arrays: a fast-convergence trilinear decomposition approach. *Multidimensional Systems and Signal Processing* **2019**, *30*, 239–256. https://doi.org/10.1007/s11045-018-0553-9.
11. Lin, Z.; Lin, M.; Champagne, B.; Zhu, W.P.; Al-Dhahir, N. Secrecy-Energy Efficient Hybrid Beamforming for Satellite-Terrestrial Integrated Networks. *IEEE Transactions on Communications* **2021**, *69*, 6345–6360. https://doi.org/10.1109/TCOMM.2021.3088898.
12. Heidenreich, P.; Zoubir, A.M.; Rubsamen, M. Joint 2-D DOA Estimation and Phase Calibration for Uniform Rectangular Arrays. *IEEE Transactions on Signal Processing* **2012**, *60*, 4683–4693. https://doi.org/10.1109/TSP.2012.2203125.
13. Ioannides, P.; Balanis, C. Uniform circular and rectangular arrays for adaptive beamforming applications. *IEEE Antennas and Wireless Propagation Letters* **2005**, *4*, 351–354. https://doi.org/10.1109/LAWP.2005.857039.
14. Singh, S. Minimal Redundancy Linear Array and Uniform Linear Arrays Beamforming Applications in 5G Smart Devices. *Emerging Science Journal* **2021**, *4*, 70–84. https://doi.org/10.28991/esj-2021-SP1-05.
15. Goodwin, M.; Elko, G. Constant beamwidth beamforming. In Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing, Minneapolis, MN, USA, 1993; pp. 169–172 vol.1. https://doi.org/10.1109/ICASSP.1993.319082.
16. Zarifi, K.; Affes, S.; Ghrayeb, A. Collaborative Null-Steering Beamforming for Uniformly Distributed Wireless Sensor Networks. *IEEE Transactions on Signal Processing* **2010**, *58*, 1889–1903. https://doi.org/10.1109/TSP.2009.2036476.
17. Frost, O. An algorithm for linearly constrained adaptive array processing. *Proceedings of the IEEE* **1972**, *60*, 926–935. https://doi.org/10.1109/PROC.1972.8817.
18. Buckley, K. Spatial/Spectral filtering with linearly constrained minimum variance beamformers. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **1987**, *35*, 249–266. https://doi.org/10.1109/TASSP.1987.1165142.
19. Souden, M.; Benesty, J.; Affes, S. A Study of the LCMV and MVDR Noise Reduction Filters. *IEEE Transactions on Signal Processing* **2010**, *58*, 4925–4935. https://doi.org/10.1109/TSP.2010.2051803.
20. Hixson, E.L.; Au, K.T. WideBandwidth ConstantBeamwidth Acoustic Array. *The Journal of the Acoustical Society of America* **1970**, *48*, 117–117. https://doi.org/10.1121/1.1974937.
21. Wang, Z.; Li, J.; Stoica, P.; Nishida, T.; Sheplak, M. Constant-beamwidth and constant-powerwidth wideband robust Capon beamformers for acoustic imaging. *The Journal of the Acoustical Society of America* **2004**, *116*, 1621–1631. https://doi.org/10.1121/1.1744751.

22. Long, T.; Cohen, I.; Berdugo, B.; Yang, Y.; Chen, J. Window-Based Constant Beamwidth Beamformer. *Sensors* **2019**, *19*, 2091. https://doi.org/10.3390/s19092091.
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, August 2022; pp. 50–54. https://doi.org/10.23919/EUSIPCO55093.2022.9909899.
24. Benesty, J.; Chen, J.; Huang, Y. *Microphone Array Signal Processing*; Vol. 1, *Springer Topics in Signal Processing*, Springer Berlin Heidelberg: Berlin, Heidelberg, 2008. https://doi.org/10.1007/978-3-540-78612-2.
25. Abramovich, Y.I.; Frazer, G.J.; Johnson, B.A. Iterative Adaptive Kronecker MIMO Radar Beamformer: Description and Convergence Analysis. *IEEE Transactions on Signal Processing* **2010**, *58*, 3681–3691. https://doi.org/10.1109/TSP.2010.2046081.
26. Werner, K.; Jansson, M.; Stoica, P. On Estimation of Covariance Matrices With Kronecker Product Structure. *IEEE Transactions on Signal Processing* **2008**, *56*, 478–491. https://doi.org/10.1109/TSP.2007.907834.
27. Habets, E.A.P.; Gannot, S. Generating sensor signals in isotropic noise fields. *The Journal of the Acoustical Society of America* **2007**, *122*, 3464–3470. https://doi.org/10.1121/1.2799929.
28. Markovich-Golan, S.; Gannot, S.; Kellermann, W. Combined LCMV-TRINICON Beamforming for Separating Multiple Speech Sources in Noisy and Reverberant Environments. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **2017**, *25*, 320–332. https://doi.org/10.1109/TASLP.2016.2633806.
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, September 2016, pp. 1981–1985. https://doi.org/10.21437/Interspeech.2016-552.
30. Kaiser, J.; Schafer, R. On the use of the I0-sinh window for spectrum analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **1980**, *28*, 105–107. https://doi.org/10.1109/TASSP.1980.1163349.
31. Brandstein, M.; Ward, D.; Lacroix, A.; Venetsanopoulos, A., Eds. *Microphone Arrays: Signal Processing Techniques and Applications*; Digital Signal Processing, Springer Berlin Heidelberg: Berlin, Heidelberg, 2001. https://doi.org/10.1007/978-3-662-04619-7.
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, May 2020; pp. 211–215. https://doi.org/10.1109/ICASSP40776.2020.9052988.
33. Hamahata, Y. Arithmetic functions and the Cauchy product. *Archiv der Mathematik* **2020**, *114*, 41–50. https://doi.org/10.1007/s00013-019-01384-9.