

Article

Window-Based Constant Beamwidth Beamformer

Tao Long ^{1,*}, Israel Cohen ², Baruch Berdugo ², Yan Yang ¹ and Jingdong Chen ¹

¹ Center of Intelligent Acoustics and Immersive Communications, Northwestern Polytechnical University, 127 Youyi West Road, Xi'an 710072, Shaanxi, China; y.yang@nwpu.edu.cn (Y.Y.); jingdongchen@ieee.org (J.C.)

² Department of Electrical Engineering, Technion—Israel Institute of Technology, Technion City, Haifa 32000, Israel; icohen@ee.technion.ac.il (I.C.); bbaruch@technion.ac.il (B.B.)

* Correspondence: longtao@nwpu.edu.cn

Received: 31 March 2019; Accepted: 26 April 2019; Published: 6 May 2019



Abstract: Beamformers have been widely used to enhance signals from a desired direction and suppress noise and interfering signals from other directions. Constant beamwidth beamformers enable a fixed beamwidth over a wide range of frequencies. Most of the existing approaches to design constant beamwidth beamformers are based on optimization algorithms with high computational complexity and are often sensitive to microphone mismatches. Other existing methods are based on adjusting the number of sensors according to the frequency, which simplify the design, but cannot control the sidelobe level. Here, we propose a window-based technique to attain the beamwidth constancy, in which different shapes of standard window functions are applied for different frequency bins as the real weighting coefficients of microphones. Thereby, not only do we keep the beamwidth constant, but we also control the sidelobe level. Simulation results show the advantages of our method compared with existing methods, including lower sidelobe level, higher directivity factor, and higher white noise gain.

Keywords: constant beamwidth beamformer; microphone arrays; chebyshev window; Kaiser window; discrete prolate spheroidal sequences

1. Introduction

Beamformers, or spatial filters, enhance signals from a desired direction and suppress noise and interfering signals from other directions. Deterministic and adaptive beamforming techniques have been widely studied and used in radar, sonar, seismology, tomography, communication and many other areas [1–5].

A variety of beamforming techniques are available, including data-independent, statistically optimal and adaptive approaches. Traditional beamforming techniques suffer from a frequency varying beamwidth, which restricts their deployment in broad-band applications (e.g., speech communication). The basic approach of solving this problem is to design a constant beamwidth beamformer, where the beampattern maintains a fixed beamwidth over a wide frequency band. Many existing methods have been investigated to obtain constant beamwidths [6–12]. These methods are mainly based on optimization algorithms with high computational design complexity, and they are often sensitive to microphone mismatches.

Recently, Rosen et al. [13] proposed finite impulse response (FIR) based beamformers [14] with constant beamwidths. The main idea behind this approach is to change the effective array aperture in each frequency bin to maintain the beamwidth constant over the desired frequency band. This method is characterized by low computational complexity, but cannot control the sidelobe level. If we regard the coefficients of the FIR filter as a window function, then the beamformer is analogous to the discrete Fourier transform of the window. It can be shown that the FIR filter proposed in [13] is a

kind of modified rectangular window, and it inspires us to use other kinds of windows. As a result, some standard window functions whose shapes are controlled by a single parameter can be directly used to obtain different beamwidths and sidelobe levels, e.g., a discrete prolate spheroidal sequences (DPSS) window can maximize the relative beamforming power that is concentrated in an angular region, and a Chebyshev window minimizes the beamwidth for a given sidelobe level.

In this paper, we propose window-based beamformers with constant beamwidths. The main idea is to apply different shapes of windows for different frequency bins as real weighting coefficients of microphones, so that the beamwidth is maintained constant by controlling the window parameters. The rest of this paper is organized as follows. In Section 2, we introduce the signal model and illustrate the frequency varying beamwidth problem for traditional uniform linear arrays. Section 3 proposes the window-based beamformer with a constant beamwidth, using modified rectangular, DPSS, Kaiser and Chebyshev windows. Section 4 evaluates the performances of different window-based methods using the white noise gain and directivity factor. Finally, some conclusions are drawn in Section 5.

2. Signal Model and Problem Formulation

2.1. Signal Model

Consider a uniform linear array (ULA) consisting of M omnidirectional microphones, with an interelement spacing equal to δ . Assume that there are an odd number ($M = 2N + 1$) of microphones, as shown in Figure 1, whose locations are:

$$x_m = m\delta, m = -N, -(N-1), \dots, N-1, N, \quad (1)$$

where m denotes the microphone index, and x_m denotes the location of the m th microphone.

We consider the farfield case where a source of interest radiates an acoustic wave that propagates in an anechoic environment at the speed of sound, i.e., $c = 340$ m/s. The direction of the source signal is parameterized by the angle θ measured with respect to the broadside of the linear array. In the frequency domain, the signal model at the frequency index f can be written as:

$$\begin{aligned} Y_m(f) &= X_m(f) + V_m(f) \\ &= e^{-jm(2\pi f\delta/c)\sin\theta} X_0(f) + V_m(f), \end{aligned} \quad (2)$$

where $Y_m(f)$, $X_m(f)$, and $V_m(f)$ are the frequency-domain representations of the received noisy signal, the desired source signal and the additive noise signal at the m th microphone, respectively, $f > 0$ is the temporal frequency, and j is the imaginary unit with $j^2 = -1$. In a vector form, we can rearrange the signal model in Equation (2) as

$$\begin{aligned} \mathbf{y}(f) &= \mathbf{x}(f) + \mathbf{v}(f) \\ &= \mathbf{d}(f, \theta) X_0(f) + \mathbf{v}(f), \end{aligned} \quad (3)$$

where

$$\begin{aligned} \mathbf{y}(f) &\triangleq \begin{bmatrix} Y_{-N}(f) & \cdots & Y_0(f) & \cdots & Y_N(f) \end{bmatrix}^T, \\ \mathbf{x}(f) &\triangleq \begin{bmatrix} X_{-N}(f) & \cdots & X_0(f) & \cdots & X_N(f) \end{bmatrix}^T, \\ \mathbf{v}(f) &\triangleq \begin{bmatrix} V_{-N}(f) & \cdots & V_0(f) & \cdots & V_N(f) \end{bmatrix}^T, \end{aligned} \quad (4)$$

the superscript T is the transpose operator, and

$$\mathbf{d}(f, \theta) \triangleq \begin{bmatrix} e^{jN(2\pi f\delta/c)\sin\theta} & \cdots & 1 & \cdots & e^{-jN(2\pi f\delta/c)\sin\theta} \end{bmatrix}^T,$$

is the signal propagation vector corresponding to θ , which is in the same form as the steering vector.

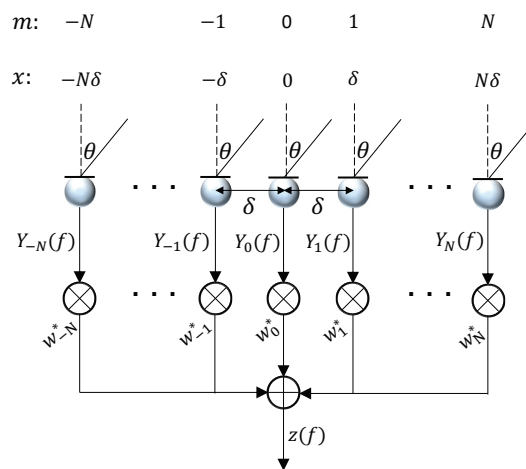


Figure 1. Beamforming with a uniform linear array consisting of M ($M = 2N + 1$) microphones with spacing δ , where m denotes the microphone index, x denotes the location of the microphone, θ is the direction of arrival angle measured with respect to the broadside of the linear array.

2.2. Beamformer

As shown in Figure 1, the beamformer estimates the desired signal by applying a spatial filter to the sensors' outputs [2], i.e.,

$$Z(f) = \sum_{m=-N}^N w_m^*(f) Y_m(f) = \mathbf{w}^H(f) \mathbf{y}(f), \quad (5)$$

where $*$ and H denote complex conjugation and conjugate-transpose operator, $Z(f)$ is an estimate of the desired signal $X_0(f)$, and

$$\mathbf{w}(f) \triangleq \left[w_{-N}(f) \quad \cdots \quad w_0(f) \quad \cdots \quad w_N(f) \right]^T \quad (6)$$

is the linear filter of length $2N + 1$.

One of the most important measures to quantify the performance of a beamformer is the so-called beampattern or directivity pattern, which describes the sensitivity of the beamformer to a plane wave impinging on the array from the direction θ . Mathematically, the beampattern is defined as

$$\mathcal{B}(f, \theta) \triangleq \mathbf{w}^H(f) \mathbf{d}(f, \theta) \quad (7)$$

$$= \sum_{m=-N}^N w_m^*(f) e^{-jm(2\pi f\delta/c) \sin \theta}. \quad (8)$$

2.3. Beamwidth

We now give an example of the uniform weighting beamformer (i.e., delay-and-sum beamformer),

$$w_m = \frac{1}{M}, \quad m = -N, -(N-1), \dots, N-1, N, \quad (9)$$

where $M = 2N + 1$. When $\theta \neq 0$, we easily get

$$|\mathcal{B}(f, \theta)| = \frac{1}{M} \left| \frac{1 - e^{-j(2\pi M f \delta / c) \sin \theta}}{1 - e^{-j(2\pi f \delta / c) \sin \theta}} \right|. \quad (10)$$

If we define the beamwidth θ_{BW} as the angle between the two lowest values at both sides of the main lobe (i.e., the beamwidth null to null), θ_{BW} can be obtained in this case:

$$\theta_{BW} = 2 \sin^{-1} \left(\frac{c}{\delta M f} \right). \quad (11)$$

This expression indicates the dependence of the beamwidth on the number of sensors M , interelement spacing δ and frequency f . One can observe that the beamwidth decreases as the frequency increases, which implies that this beamforming method suffers from a frequency varying beamwidth. The beampattern based on the delay-and-sum beamformer is shown in Figure 2. One observes that the beamwidth decreases as the frequency increases.

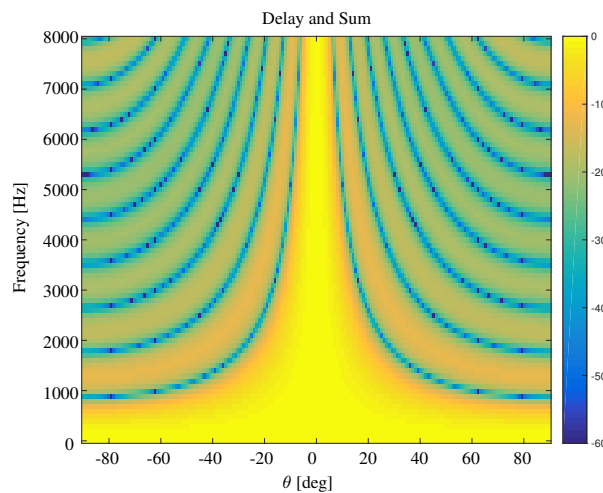


Figure 2. Beampattern using delay-and-sum beamformer. $M = 11$, $\delta = 3.5$ cm.

3. Window-Based Acoustic Beamformer with a Constant Beamwidth

In this section, we propose a window-based symmetrical beamformer method with a constant beamwidth θ_{CBW} over a wide frequency range.

Define $u \triangleq (2\pi f \delta / c) \sin \theta$, then the steering vector can be rewritten as:

$$\mathbf{d}(u) = \left[e^{jNu} \quad \dots \quad 1 \quad \dots \quad e^{-jNu} \right]^T. \quad (12)$$

Accordingly, the beampattern is obtained through the discrete Fourier transform of the spatial filter:

$$\mathcal{B}(u) = \sum_{m=-N}^N w_m^* e^{-jmu} = \mathbf{w}^H \mathbf{d}(u) \quad (13)$$

In this work, we restrict ourselves to real weights, then the beampattern is given by

$$\mathcal{B}(u) = \sum_{m=-N}^N w_m e^{-jmu} = \mathbf{w}^T \mathbf{d}(u) \quad (14)$$

The real weights w_m are regarded as a spatial window function. Hence, the beamwidth can be maintained by applying different shapes of windows for different frequency bins. Next, we present four different kinds of windows, i.e., modified rectangular, DPSS, Kaiser, Chebyshev. For each window, we first introduce the mathematical representation, and then show how to control its shape as a function of frequency by setting the proper parameter of the window in order to maintain constant beamwidth.

3.1. Modified Rectangular Window

From Equation (11), in order to keep the beamwidth constant for varying frequency, the product Mf should remain constant, which means that the number of microphones should decrease as the frequency increases. To alleviate the beamwidth fluctuations, Rosen et al. [13] proposed a modified rectangular window based on smoothing coefficients.

The weights w_m in [13] can be regarded as a kind of modified rectangular window:

$$w_m = \begin{cases} 1, & -K < m < K \\ g, & m = -K, m = K, (0 < g < 1) \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

where $2K + 1$ is the number of activated microphones, and g is the smoothing coefficient.

Next, we show how to obtain the parameter K and g to keep the beamwidth θ_{BW} constant.

3.1.1. Lowest and Highest Frequencies

Since the number of activated microphones should be less than M and more than 3, we should first determine the lowest frequency f_L and the highest frequency f_H for which the desired beamwidth is feasible. For the given array configuration and fixed beamwidth θ_{CBW} , based on Equation (11), the lowest and highest frequencies using rectangular window are

$$\begin{aligned} f_{L,\text{Rec}} &= \frac{c}{M\delta \sin(\theta_{\text{CBW}}/2)}, \\ f_{H,\text{Rec}} &= \frac{c}{3\delta \sin(\theta_{\text{CBW}}/2)}. \end{aligned} \quad (16)$$

Meanwhile, in order to avoid maximum grating lobe, the highest frequency $f_{H,\text{Rec}}$ should also be smaller than c/δ .

3.1.2. The Parameter K

For the feasible frequency range $f_L < f < f_H$, we need to reduce the number of activated microphones to keep $(2K + 1)f$ constant as the frequency varies. So the value of K is obtained by the maximum integer which satisfied $(2K + 1)f \leq Mf_L$.

3.1.3. The Parameter g

The smoothing coefficient g can be derived as follows. The beampattern using the modified rectangular window w_m is given by

$$\begin{aligned} \mathcal{B}(f, \theta) &= \sum_{m=-N}^N w_m e^{-jm(2\pi f\delta/c) \sin \theta} \\ &= 2g \cos(2K\pi f\delta \sin \theta / c) + \sum_{m=-(K-1)}^{K-1} e^{-jm(2\pi f\delta/c) \sin \theta}. \end{aligned} \quad (17)$$

The optimal value of smoothing coefficient g is obtained by setting $\mathcal{B}(f, \theta_{\text{CBW}}/2) = 0$, which yields

$$g = \frac{\sum_{m=-(K-1)}^{K-1} e^{-jm(2\pi f\delta/c) \sin(\theta_{\text{CBW}}/2)}}{-2 \cos(2K\pi f\delta \sin(\theta_{\text{CBW}}/2) / c)}. \quad (18)$$

In order to normalize the beampattern, we use the normalized weighting coefficients as:

$$w'_m = \frac{w_m}{\sum_{m=-N}^N w_m} = \frac{w_m}{2g + 2K + 1}. \quad (19)$$

The beampattern based on a modified rectangular window is shown in Figure 3. The beamwidth is fixed to 40° , $M = 11$ and $\delta = 3.5$ cm. We show the beampattern in three different frequencies $f = 4000, 5000, 6000$ Hz. One observes that the rectangular window-based method can effectively fix the beamwidth, but the sidelobe level is high in this case.

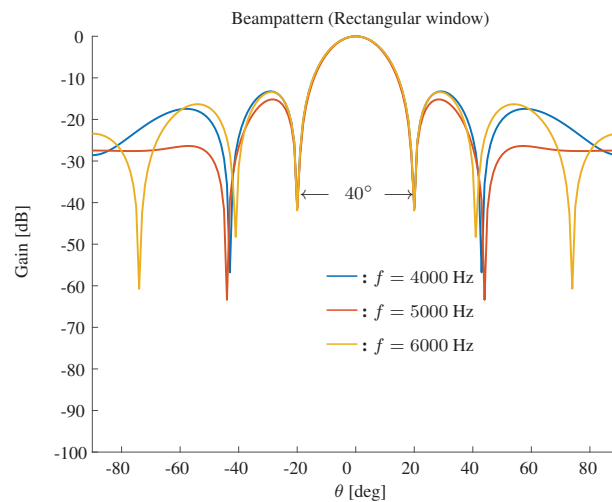


Figure 3. Constant beamwidth beampattern using a modified rectangular window. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

3.2. DPSS Window

Rectangular window-based method can effectively fix the beamwidth but cannot control the sidelobes. So, we attempt to find some other windows with real weights in order to control the sidelobe level. Based on Equation (13), we can first define α as the ratio of the total beamforming power that is concentrated in a given angular region:

$$\begin{aligned} \alpha &= \frac{\int_{-u_0}^{u_0} |\mathcal{B}(u)|^2 du}{\int_{-\pi}^{\pi} |\mathcal{B}(u)|^2 du} \\ &= \frac{\mathbf{w}^H \left[\int_{-u_0}^{u_0} \mathbf{d}(u) \mathbf{d}(u)^H du \right] \mathbf{w}}{\mathbf{w}^H \left[\int_{-\pi}^{\pi} \mathbf{d}(u) \mathbf{d}(u)^H du \right] \mathbf{w}} \\ &= \frac{\mathbf{w}^H \mathbf{A} \mathbf{w}}{\mathbf{w}^H \mathbf{B} \mathbf{w}}, \end{aligned} \quad (20)$$

where $u_0 = (2\pi f \delta / c) \sin \theta_0$, $\mathbf{A} = \int_{-u_0}^{u_0} \mathbf{d}(u) \mathbf{d}(u)^H du$, the (m, n) th element of \mathbf{A} is

$$\int_{-u_0}^{u_0} e^{jmu} e^{-jnu} du = \frac{2 \sin[(m-n)u_0]}{m-n}, \quad (21)$$

and similarly,

$$\mathbf{B} = \int_{-\pi}^{\pi} \mathbf{d}(u) \mathbf{d}(u)^H du = 2\pi \mathbf{I}, \quad (22)$$

where \mathbf{I} is the $M \times M$ identity matrix. To maximize α , the optimum solution is obtained from the eigenvalue problem

$$\mathbf{A}\mathbf{w} = \pi\lambda\mathbf{w} \quad (23)$$

or equivalently,

$$\sum_{n=-N}^N \frac{\sin[(m-n)u_0]}{m-n} w_n = \pi\lambda w_m, \quad (24)$$

where λ is an eigenvalue of \mathbf{A} . Thus in this case, α is maximized by the maximum eigenvalue λ_{\max} . The resulting weight sequences w_m are called discrete prolate spheroidal sequences [15] (DPSS, or Slepian sequences).

3.2.1. Lowest and Highest Frequencies

When $u_0 = 0$, the DPSS window becomes a rectangular window. From Equation (11), we get the lowest frequency $f_{L,DPSS}$ which enables the desired beamwidth:

$$f_{L,DPSS} = \frac{c}{M\delta \sin(\theta_{CBW}/2)}, \quad (25)$$

On the other side, in order to avoid maximum grating lobe, the highest frequency is given by

$$f_{H,DPSS} = \frac{c}{\delta}. \quad (26)$$

3.2.2. The Parameter u_0

For the given array configuration, it is easy to verify that the beamwidth increases as the parameter u_0 increases. For the given constant beamwidth θ_{CBW} , we can set for the frequency f

$$u_0 \approx (2\pi f \delta / c) \sin(\theta_{CBW}/2),$$

or we can search the optimal parameter u_0 to fix the beamwidth in practice, then w_m can be obtained following the DPSS window in Equation (24).

At last, in order to normalize the beampattern, we use the normalized weighting coefficients as:

$$w'_m = \frac{w_m}{\sum_{m=-N}^N w_m}. \quad (27)$$

The beampattern based on a DPSS window is shown in Figure 4. The beamwidth is still fixed to 40° , $M = 11$ and $\delta = 3.5$ cm. We show the beampattern in three different frequencies $f = 4000, 5000, 6000$ Hz. It can be seen that the DPSS window based approach cannot only keep the beamwidth constant, but also effectively suppresses the sidelobe level.

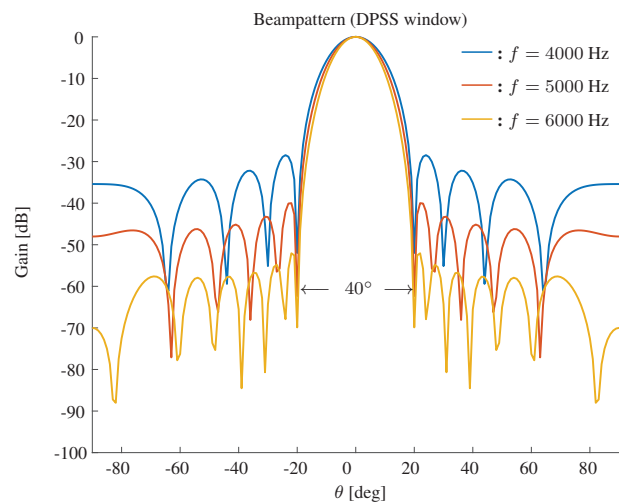


Figure 4. Constant beamwidth beampattern using a DPSS window. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

3.3. Kaiser Window

A Kaiser window is a simple approximation to the DPSS window using Bessel functions. The details of calculating a Kaiser window can be found in [16]. The weighting coefficient of the microphone with index m is:

$$w_m = \frac{J_0\left(\beta\sqrt{1 - \left(\frac{m}{N}\right)^2}\right)}{J_0(\beta)}, \quad -N \leq m \leq N \quad (28)$$

where J_0 is the zeroth-order modified Bessel function of the first kind. The parameter $\beta \geq 0$ specifies a beampattern tradeoff between the sidelobe amplitude A_{SL} and the main lobe width. When $\beta = 0$, the Kaiser window becomes a rectangular window.

3.3.1. Lowest and Highest Frequencies

Since the Kaiser window approximates the DPSS window, in order to make the desired beamwidth feasible, the lowest and highest frequencies using kaiser window are

$$\begin{aligned} f_{L,Kaiser} &= \frac{c}{M\delta \sin(\theta_{CBW}/2)}, \\ f_{H,Kaiser} &= \frac{c}{\delta}. \end{aligned} \quad (29)$$

3.3.2. The Parameter β

For the given array configuration, it is easy to verify that the beamwidth increases as the parameter β increases. For a given constant beamwidth θ_{CBW} , the parameter β can be obtained following the approximate piecewise relation [16]:

$$\begin{cases} \beta \approx 0.76608 (A_{SL} - 13.26)^{0.4} + 0.09834 (A_{SL} - 13.26), \\ A_{SL} \approx (26Mf\delta/c) \sin(\theta_{CBW}/2) - 12. \end{cases}$$

In practice, we can also search the optimal parameter β for a constant beamwidth. Furthermore, in order to normalize the beampattern, we use the normalized weighting coefficients as:

$$w'_m = \frac{w_m}{\sum_{m=-N}^N w_m}. \quad (30)$$

The beampattern based on a Kaiser window is shown in Figure 5. We fix the beamwidth to 40° , $M = 11$ and $\delta = 3.5$ cm, and plot the beampattern in three different frequencies $f = 4000, 5000, 6000$ Hz. It is shown that the Kaiser window based method can get a similar beampattern as the DPSS window.

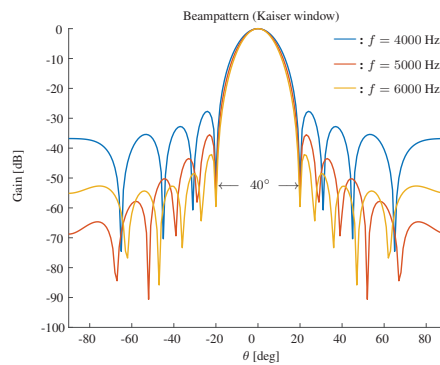


Figure 5. Constant beamwidth beampattern using a Kaiser window. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

3.4. Chebyshev Window

Another window that can be used to control the main lobe beamwidth and sidelobe level is the Chebyshev Window [17,18], which minimizes the beamwidth for a given maximum sidelobe level. The coefficients w_m of the Chebyshev window are given by

$$w_m = \frac{1}{M} \left[1 + 2r \sum_{n=1}^N T_{2N} \left(x_0 \cos \left(\frac{n\pi}{M} \right) \right) \cos \left(\frac{2\pi nm}{M} \right) \right], \quad (31)$$

where $-N \leq m \leq N$, $M = 2N + 1$, $x_0 = \cosh \left(\frac{1}{2N} \cosh^{-1} \left(\frac{1}{r} \right) \right)$, r is defined as the amplitude ratio between maximum sidelobe and mainlobe, and $T_m(x)$ is the Chebyshev polynomial of the first kind, defined by

$$T_m(x) = \begin{cases} \cos \left(m \cos^{-1}(x) \right) & |x| \leq 1 \\ \cosh \left(m \cosh^{-1}(x) \right) & |x| > 1 \end{cases} \quad (32)$$

3.4.1. Lowest and Highest Frequencies

When the sidelobe attenuation is the same as the mainlobe, or $r = 1$, the coefficients w_m of the Chebyshev window are

$$w_m = \begin{cases} 1, & m = -N, m = N \\ 0, & \text{otherwise.} \end{cases} \quad (33)$$

It is equivalent to a two elements array with an interelement spacing $2N\delta$. According to Equations (25) and (26), the lowest and highest frequencies using Chebyshev window are:

$$\begin{aligned} f_{L,\text{Cheb}} &= \frac{c}{4N\delta \sin(\theta_{\text{CBW}}/2)}, \\ f_{H,\text{Cheb}} &= \frac{c}{\delta}. \end{aligned} \quad (34)$$

3.4.2. The Parameter r

For the given array configuration, it is easy to verify that the beamwidth increases as the parameter r decreases. For a given constant beamwidth θ_{CBW} , the parameter r can be obtained following the approximate piecewise relation:

$$\begin{cases} r &= 1 / \cosh \left(2N \cosh^{-1} x_0 \right), \\ x_0 &\approx 1 / \cos \left((\pi f \delta / c) \sin \left(\frac{\theta_{\text{CBW}}}{2} \right) \right). \end{cases}$$

In practice, we can also search the optimal parameter r for a constant beamwidth. Again, in order to normalize the beampattern, we use the normalized weighting coefficients as:

$$w'_m = \frac{w_m}{\sum_{m=-N}^N w_m}. \quad (35)$$

The beampatterns for different frequencies based on a Chebyshev window are shown in Figure 6. Again, the beamwidth is fixed to 40° , and one can find that the Chebyshev window based method can also effectively keep the beamwidth constant and yield equi-level sidelobes.

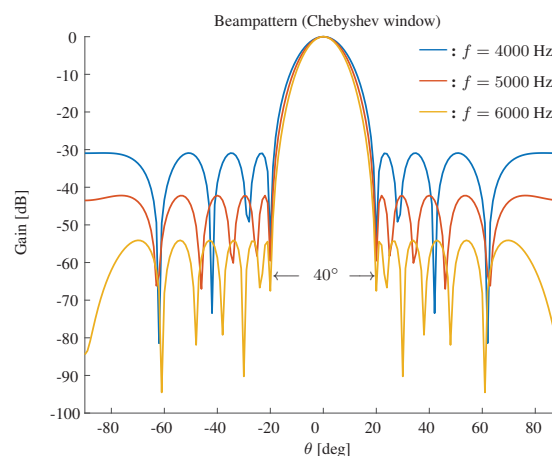


Figure 6. Constant beamwidth beampattern using the Chebyshev window. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

4. Experiments and Results

We have discussed different window-based beamformers with a constant beamwidth in Section 3. In this section, we compare the performances of different windows via several simulations. All of the simulated uniform linear arrays are configured with $M = 11$ omnidirectional microphones, with an interelement spacing equal to $\delta = 3.5$. In these experiments, the modified rectangular, DPSS, Kaiser and Chebyshev window-based methods are used and the beamwidth is fixed to $\theta_{\text{CBW}} = 40^\circ$. In Section 4.1 we show how to set the optimal parameter to shape the window and design the beamformer in order to attain beamwidth constancy. Section 4.2 provides the performance measures of the beamformer. Section 4.3 shows the results.

4.1. Optimal Window Parameter

In Section 3, we have shown how to set the parameter to shape the window in order to attain beamwidth constancy, e.g., the parameter g for the modified rectangular window, the parameter u_0 for the DPSS window, the parameter β for the Kaiser window, and the parameter r for the Chebyshev window. Compared with the optimization-based method, one can see that our method has very low

computational complexity in the design process, because the problem has been simplified as obtaining a single parameter of the standard window.

In practice, we can also search the optimal parameter of a given window. There are two reasons for using the search method: (1) The relationship between the fixed beamwidth and the parameter is approximate piecewise in some cases. (2) The search method can be easily extended to non uniform arrays. Algorithm 1 shows the search algorithm of a window-based beamformer with a constant beamwidth, where we search the optimal parameter of a given window for each frequency bin to keep the beamwidth fixed to $\theta_{CBW} = 40^\circ$, f_L and f_H are the lower and upper cutoff frequencies of the frequency band, respectively.

Algorithm 1 Algorithm for Searching the Optimal Parameter.

```

for  $f = f_L : f_H$ 
  Initial the parameter of window  $\mathbf{w}$ 
  Search  $\theta_{\min}$  for the first lowest value of  $|\mathcal{B}(f, \theta)| = |\mathbf{w}^H(f) \mathbf{d}(f, \theta)|$ 
  while  $\theta_{\min} \neq \theta_{CBW}/2$ 
    Increase (or decrease) the parameter of window  $\mathbf{w}$ 
    Search  $\theta_{\min}$  for the first lowest value of  $|\mathcal{B}(f, \theta)| = |\mathbf{w}^H(f) \mathbf{d}(f, \theta)|$ 
  end
end

```

4.2. Performance Measures

We evaluate the beamformers using white noise gain (WNG) and directivity factor (DF) [1,4,19]. The WNG is a measure indicating the array gain in the presence of uncorrelated white noise, which is also a measure of the sensitivity of the microphone array to some of its imperfections, such as sensor noise and mismatch. The DF of the array is the gain in signal-to-noise ratio (SNR) for the case of spherical diffuse noise. Mathematically, they are respectively defined as \mathcal{W} and \mathcal{D} (note that the main lobe is perpendicular to the line that connects all the array elements):

$$\mathcal{W}[\mathbf{w}(f)] = \frac{|\mathbf{w}^H(f) \mathbf{d}(f, 0)|^2}{\mathbf{w}^H(f) \mathbf{w}(f)}, \quad (36)$$

$$\begin{aligned} \mathcal{D}[\mathbf{w}(f)] &= \frac{|\mathcal{B}[\mathbf{w}(f), 0]|^2}{\frac{1}{2} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} |\mathcal{B}[\mathbf{w}(f), \theta]|^2 \sin \theta d\theta} \\ &= \frac{|\mathbf{w}^H(f) \mathbf{d}(f, 0)|^2}{\frac{1}{2} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \mathbf{w}^H(f) \mathbf{d}(f, \theta) \mathbf{d}(f, \theta)^H \mathbf{w}(f) \sin \theta d\theta}. \end{aligned} \quad (37)$$

We also evaluate the beamformers beamwidth which is defined as the angle between the two lowest values at both sides of the main lobe (i.e., the beamwidth null to null).

4.3. Results

4.3.1. Wideband Beampatterns with a Constant Beamwidth Using Different Windows

We first compare the constant beamwidth beampatterns using different kinds of windows, where the frequency range is $0 < f < 8000$ Hz. The results are plotted in Figure 7 for the modified rectangular window, Figure 8 for the DPSS window, Figure 9 for the Kaiser window, and Figure 10 for the Chebyshev window. One can see that all the window-based methods can effectively keep the beamwidth constant over a wide frequency band. Compared with modified rectangular window [13], the other proposed windows can obtain much lower sidelobe levels.

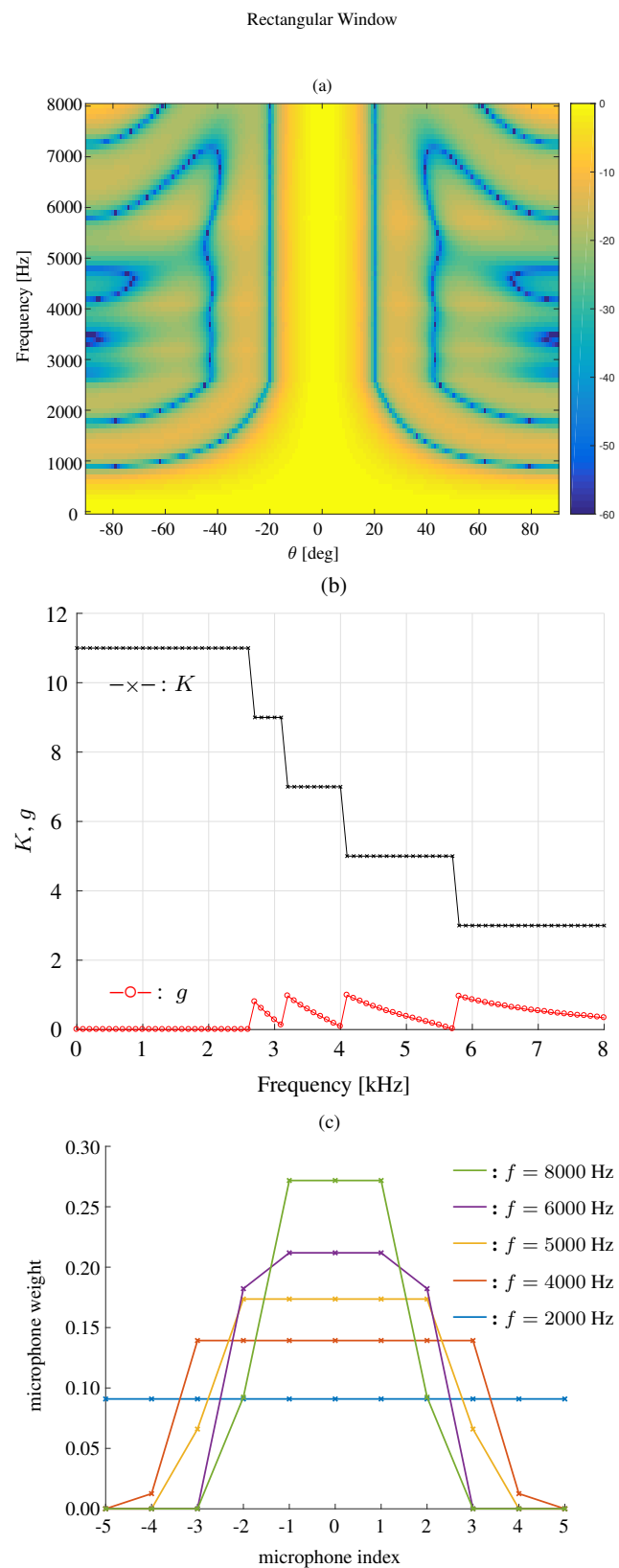


Figure 7. Constant beamwidth beamforming using the modified rectangular window-based method: (a) constant beamwidth beampattern, (b) the parameter K, g , (c) the weights of microphones for different frequency bins. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

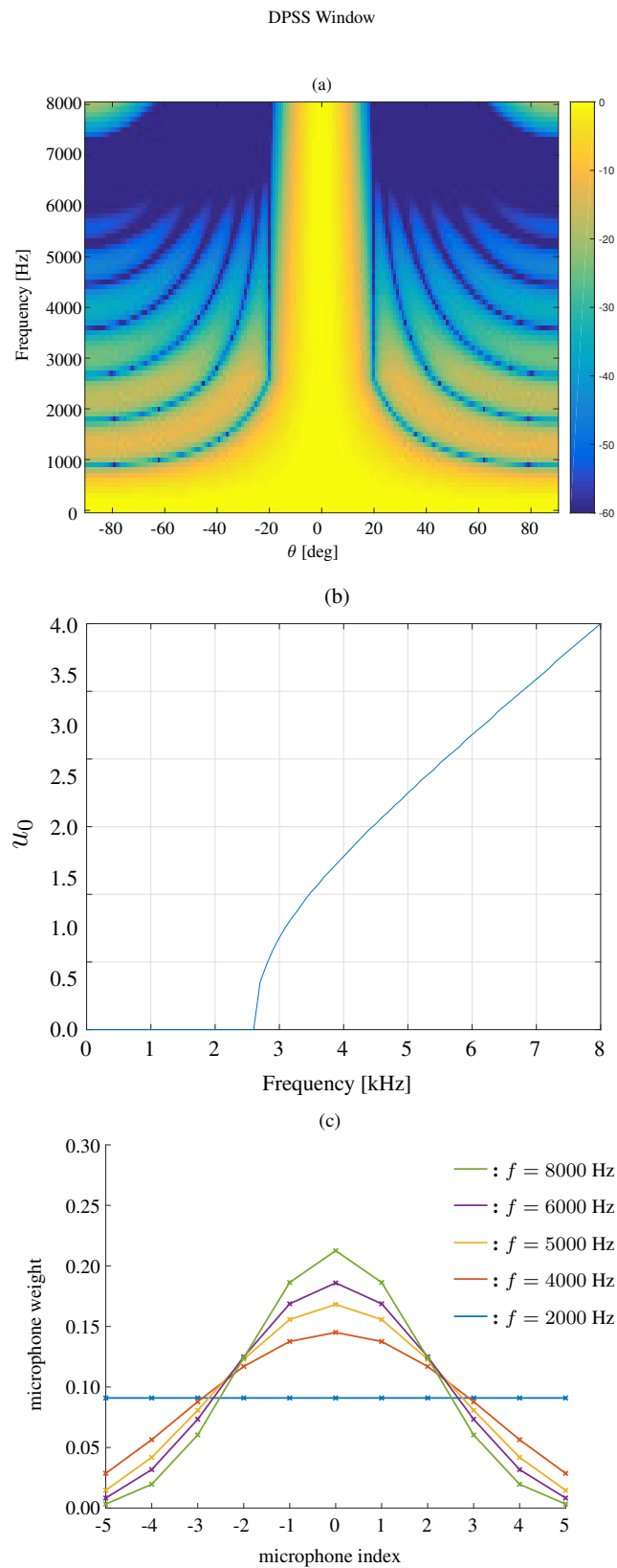


Figure 8. Constant beamwidth beamforming using the DPSS window-based method: (a) constant beamwidth beampattern, (b) the parameter u_0 , (c) the weights of microphones for different frequency bins. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

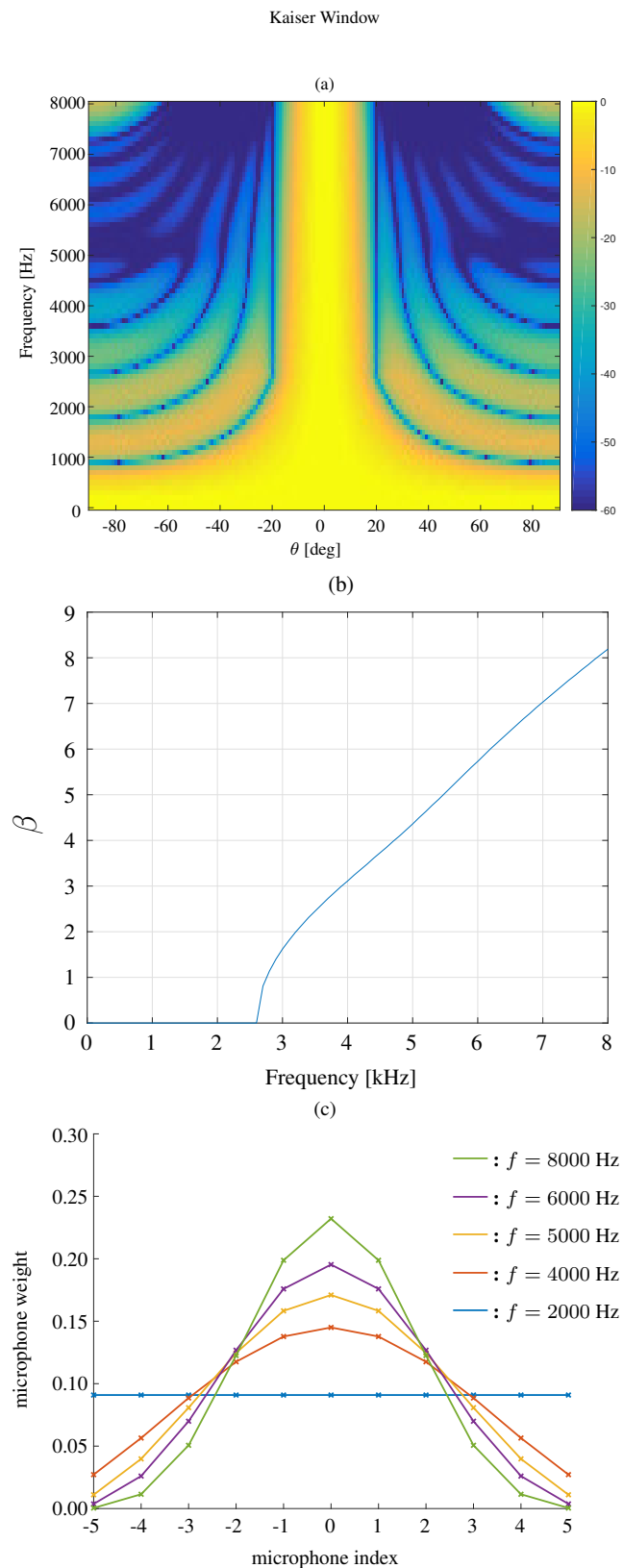


Figure 9. Constant beamwidth beamforming using the Kaiser window-based method: (a) constant beamwidth beampattern, (b) the parameter β , (c) the weights of microphones for different frequency bins. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

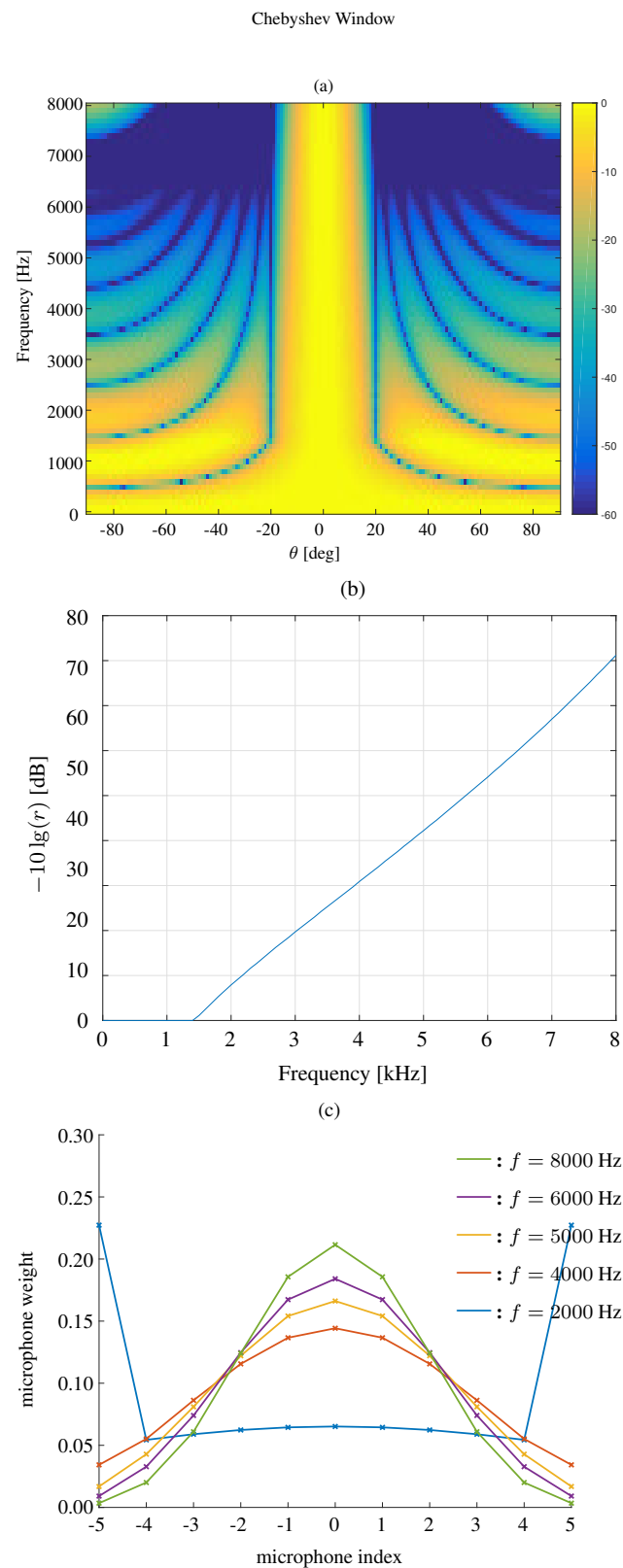


Figure 10. Constant beamwidth beamforming using the Chebyshev window-based method: (a) constant beamwidth beampattern, (b) the parameter r , (c) the weights of microphones for different frequency bins. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

4.3.2. Chebyshev Window

In order to explain the performance based on the Chebyshev window in low frequency bins, we show the weights of the microphones using the Chebyshev window when $f = 1000, 1500, 2000, 2500, 3000$ Hz, and the results are plotted in Figure 11. One can find that the Chebyshev approach designs a kind of 'saddle' shape window, where high weights are set for the microphones at the edges of the array. Suppose an extreme situation that we only use the two microphones at the edge, which means the Chebyshev window will be

$$\mathbf{w} = \left[0.5 \ 0 \ \dots \ 0 \ \dots \ 0 \ 0.5 \right]^T, \quad (38)$$

the interelement spacing will be $\delta' = 2N\delta$ and the number of microphones is $M' = 2$. Based on Equation (11), the lowest frequency $f_{L,\text{Cheb}}$ which can attain the fixed beamwidth θ_{CBW} is

$$\begin{aligned} f_{L,\text{Cheb}} &= \frac{c}{M'\delta' \sin(\theta_{\text{CBW}}/2)} = \frac{c}{4N\delta \sin(\theta_{\text{CBW}}/2)} \\ &< \frac{c}{M\delta \sin(\theta_{\text{CBW}}/2)} = f_{L,\text{Rec}} = f_{L,\text{DPSS}} = f_{L,\text{Kaiser}}. \end{aligned} \quad (39)$$

With this method, it is equivalent to increasing the interelement spacing and the virtual length of the array. As a result, the Chebyshev window-based beamformer can attain beamwidth constancy in lower frequencies compared with the other windows-based beamformers.

In order to improve the DF using a Chebyshev window in low frequency bins, we can search the parameter r to reach a compromise between beamwidth and DF. We call this compromised method a Chebyshev window-I. The search criteria in this case will increase (or decrease) the parameter of the window to maximize the directivity index, given the beamwidth is above the desired minimal beamwidth. Figure 12 shows the weights of microphones using Chebyshev window-I when $f = 1000, 1500, 2000, 2500, 3000$. The beampatterns and the parameters as functions of frequency using this method are shown in Figure 13.

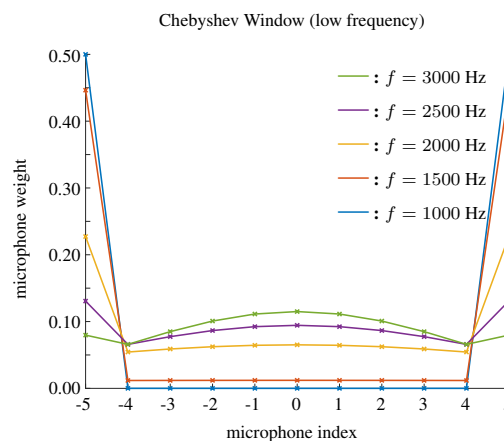


Figure 11. The weights of microphones for low frequency bins using the Chebyshev window, $M = 11$, $\delta = 3.5$ cm.

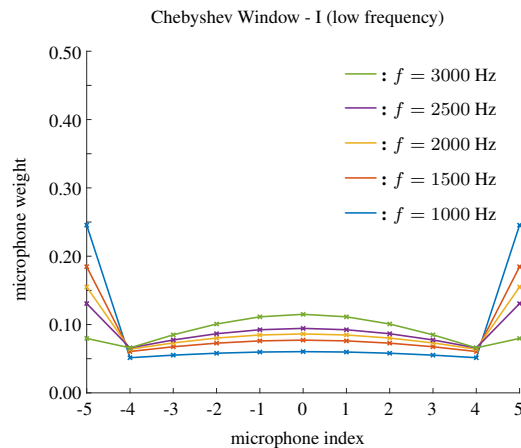


Figure 12. The weights of microphones for low frequency bins using the Chebyshev window-I, $M = 11$, $\delta = 3.5$ cm.

4.3.3. Directivity Factor and White Noise Gain as Function of Frequency

The DF and WNG as functions of frequency for different window-based beamformer are plotted in Figures 14 and 15. It is shown that the DPSS, Kaiser and Chebyshev window-based beamformer can achieve higher WNG and DF compared with the rectangular window in high frequencies. For frequencies below 2500 Hz, one can also see that the Chebyshev window-I beamformer gets highest DF compared with the other beamformers.

4.3.4. Beamwidth as Function of Frequency

At last, we compare the beamwidth as function of frequency for different window-based beamformers, and the results are plotted in Figure 16. For high frequencies ($2500 < f < 8000$ Hz), all the window-based beamformers can effectively fix the beamwidth to be 40° . For frequencies below 2500 Hz, it can be found that the Chebyshev window-based beamformer obtains smaller beamwidth compared with the other beamformers. As a result, the Chebyshev window-based beamformer can reach to the fixed beamwidth (40°) at nearly 1400 Hz, but the other window-based beamformers attain beamwidth constancy only above 2500 Hz.

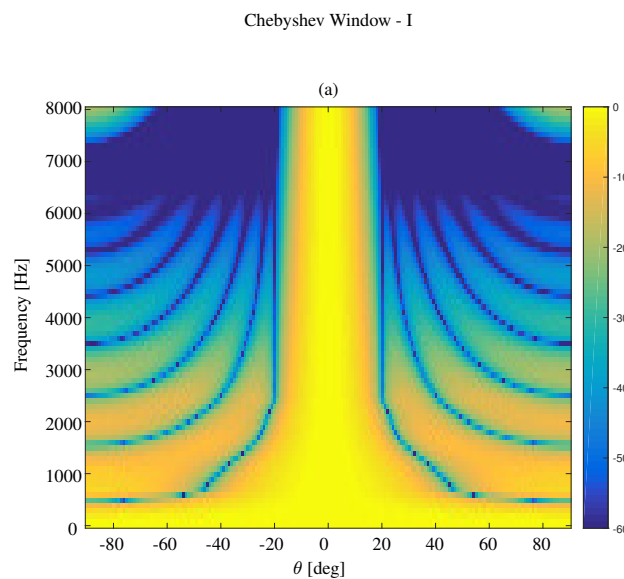


Figure 13. Cont.

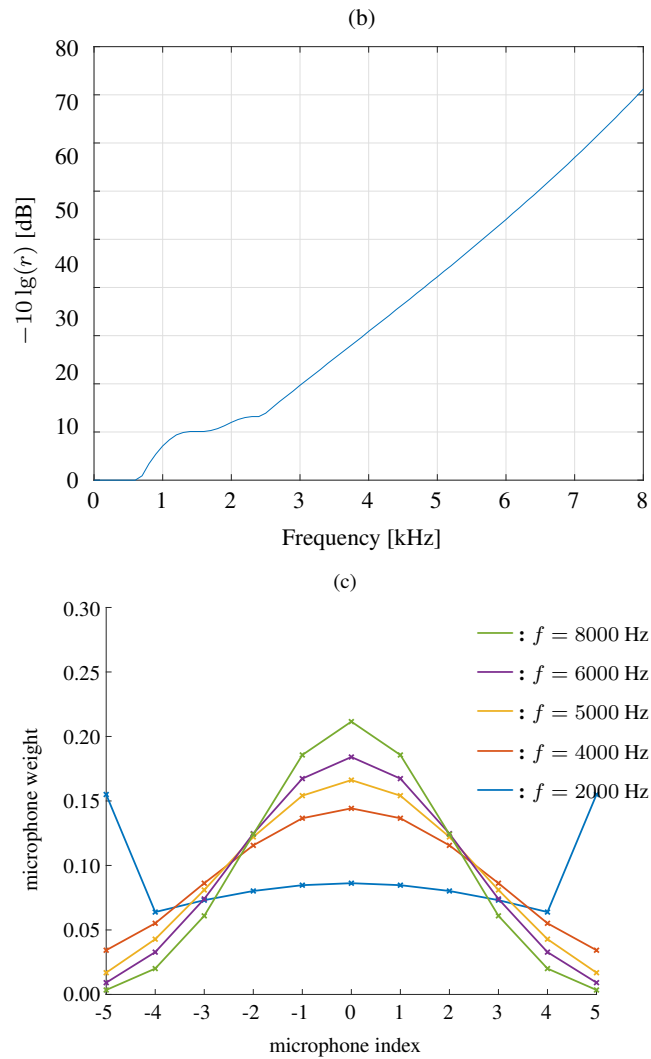


Figure 13. Constant beamwidth beamforming using the Chebyshev window-I method: (a) constant beamwidth beampattern, (b) the parameter r , (c) the weights of microphones for different frequency bins. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

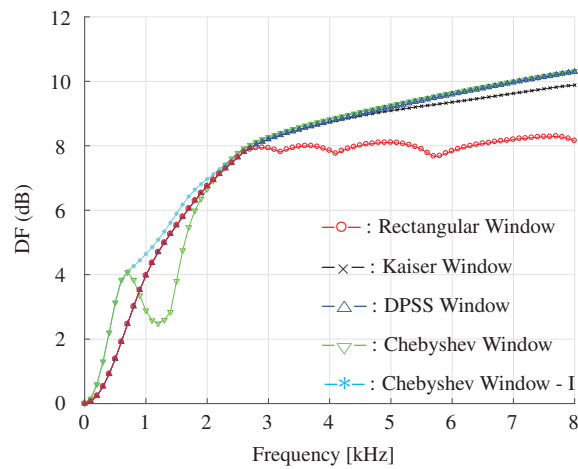


Figure 14. DF as a function of frequency for different window-based beamformers with a constant beamwidth. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

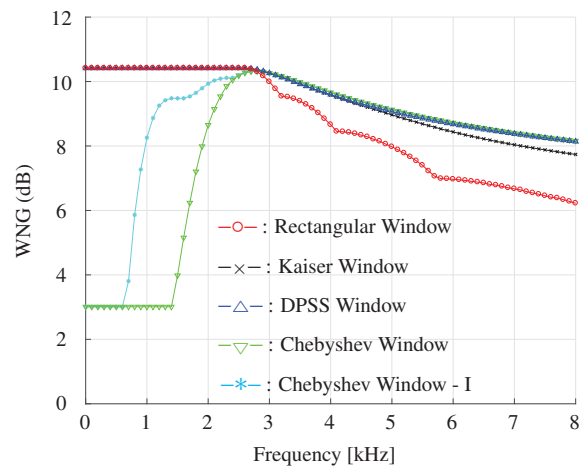


Figure 15. WNG as a function of frequency for different window-based beamformers with a constant beamwidth. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

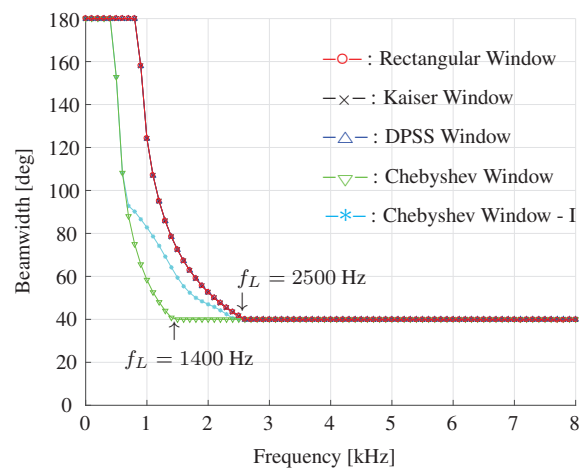


Figure 16. Beamwidth as a function of frequency for different window-based beamformers. The beamwidth is fixed to 40° , $M = 11$, $\delta = 3.5$ cm.

5. Conclusions

Traditional beamforming techniques suffer from a frequency varying beamwidth, which restricts their deployment in broadband applications. We have proposed window-based beamformers with constant beamwidths. Our method can effectively fix the beamwidth and exhibits the following advantages: (1) Compared with the optimization-based method, the proposed window-based approach is characterized by lower computational design complexity and higher white noise gain (which means it is less sensitive to microphone mismatches). (2) Compared with the FIR-based method proposed in [13], our window-based approach can reduce the sidelobe level and obtain higher directivity factor. Experiments corroborate the theoretical analysis and show that we can adjust the parameter of window to get the tradeoff between WNG and DF. Furthermore, hybrid window based beamformer design method is a topic for future research, which facilitates different windows for different frequency bins.

Author Contributions: Conceptualization, T.L. and I.C.; methodology, T.L. and J.C.; software, T.L.; validation, B.B.; formal analysis, T.L.; investigation, T.L.; resources, I.C. and J.C.; data curation, T.L.; writing—original draft preparation, T.L.; writing—review and editing, I.C., B.B. and J.C.; visualization, T.L. and Y.Y.; supervision, J.C. and I.C.; project administration, J.C. and I.C.; funding acquisition, J.C. and I.C.

Funding: This work was supported in part by the NSFC for Young Scientists under Grant 61601370, the NSFC Distinguished Young Scientists Fund under Grant 61425005 and NSFC Key Program under Grant 61831019. This work was also supported in part by the Israel Science Foundation (grant no. 576/16), the ISF-NSFC joint research program (grant No. 2514/17 and 61761146001), and Top International University Visiting Program for Outstanding Young scholars of Northwestern Polytechnical University.

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

References

1. Benesty, J.; Cohen, I.; Chen, J. *Fundamentals of Signal Enhancement and Array Signal Processing*; Wiley-IEEE Press: Singapore, 2018.
2. Benesty, J.; Chen, J.; Huang, Y. *Microphone Array Processing*; Springer: Berlin, Germany, 2008.
3. Van Trees, H.L. *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*; John Wiley & Sons: New York, NY, USA, 2004.
4. Benesty, J.; Chen, J.; Cohen, I. *Design of Circular Differential Microphone Arrays*; Springer: Cham, Switzerland, 2015.
5. Darsena, D.; Verde, F. Minimum-Mean-Output-Energy Blind Adaptive Channel Shortening for Multicarrier SIMO Transceivers. *IEEE Trans. Signal Process.* **2007**, *55*, 5755–5771. [[CrossRef](#)]
6. Ward, D.B.; Kennedy, R.A.; Williamson, R.C. FIR filter design for frequency invariant beamformers. *IEEE Signal Process. Lett.* **1996**, *3*, 69–71. [[CrossRef](#)]
7. Parra, L.C. Steerable frequency-invariant beamforming for arbitrary arrays. *J. Acoust. Soc. Am.* **2006**, *119*, 3839–3847. [[CrossRef](#)]
8. Liu, W.; Weiss, S.; McWhirter, J.G.; Proudler, I.K. Frequency invariant beamforming for two-dimensional and three-dimensional arrays. *Signal Process.* **2007**, *87*, 2535–2543. [[CrossRef](#)]
9. Markovich, S.; Gannot, S.; Cohen, I. Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals. *IEEE Trans. Audio Speech Lang. Process.* **2009**, *17*, 1071–1086. [[CrossRef](#)]
10. Crocco, M.; Trucco, A. Design of robust superdirective arrays with a tunable tradeoff between directivity and frequency-invariance. *IEEE Trans. Signal Process.* **2011**, *59*, 2169–2181. [[CrossRef](#)]
11. Tourbabin, V.; Agmon, M.; Rafaely, B.; Tabrikian, J. Optimal real-weighted beamforming with application to linear and spherical arrays. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *20*, 2575–2585. [[CrossRef](#)]
12. Wu, Y.I.; Wong, K.T.; Yuan, X.; Lau, S.K.; Tang, S.K. A directionally tunable but frequency-invariant beamformer on an acoustic velocity-sensor triad to enhance speech perception. *J. Acoust. Soc. Am.* **2012**, *131*, 3891–3902. [[CrossRef](#)] [[PubMed](#)]
13. Rosen, O.; Cohen, I.; Malah, D. FIR-based symmetrical acoustic beamformer with a constant beamwidth. *Signal Process.* **2017**, *130*, 365–376. [[CrossRef](#)]
14. Ward, D.B.; Williamson, R.C.; Kennedy, R.A. Broadband microphone arrays for speech acquisition. *Acoust. Aust.* **1998**, *26*, 17–20.
15. Slepian, D.; Pollak, H.O. Prolate spheroidal wave functions, Fourier analysis and uncertainty. *Bell Labs Tech. J.* **1961**, *40*, 43–63. [[CrossRef](#)]
16. Kaiser, J.F. Nonrecursive digital filter design using the I_0 -sinh window function. In Proceedings of the IEEE International Symposium on Circuits & Systems, San Francisco, CA, USA, April 1974.
17. Dolph, C. A current distribution for broadside arrays which optimizes the relationship between beam width and side-lobe level. *Proc. IRE* **1946**, *34*, 335–348. [[CrossRef](#)]
18. Lyons, R.G. *Understanding Digital Signal Processing*, 3rd ed.; Pearson Education: Delhi, India, 2011.
19. Pan, C.; Chen, J.; Benesty, J. Theoretical analysis of differential microphone array beamforming and an improved solution. *IEEE Trans. Audio Speech Lang. Process.* **2015**, *23*, 2093–2105. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).