

Phased microphone planar arrays

Lucas Åkerstedt, Mika Söderström

August 12, 2022

1 Introduction

By creating an antenna consisting of multiple antenna elements, it is possible to radiate in a desired direction by electrically introducing a phase shift to each antenna element. This type of antenna is called a phased array antenna. If the elements are microphones, it can therefore be possible to create a microphone array to spatially filter out signals from a desired direction, or identify direction of arrival of sound waves from one or several different sources.

In this report, the geometry and performance of planar arrays are discussed in general. Then, a beam forming algorithm for wide band signals in the time domain is described. The algorithm is evaluated with both emulated data as well as measured data with a real acoustic array. Lastly, some future developments are proposed as a continuation of this project.

2 Planar array geometry

In section, we want to show how the geometry of a phased array antenna can alter its performance.

2.1 Theory

When analyzing the behavior of an antenna, we often need the far-field of the antenna. To obtain the far-field of an antenna, we can utilize the far-field approximation. In this case, we are only using a sum of antenna elements.

We start by deriving the far-field behavior for an isotropic antenna element. Such an element would radiate a spherical wave:

$$\frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}, \quad (1)$$

where \mathbf{r} is the field vector and \mathbf{r}' is the source vector. By assuming a large distance from the antenna ($kr \gg 1$ and $r \gg r'$), we can approximate our spherical wave as following:

$$|\mathbf{r} - \mathbf{r}'| = \sqrt{(\mathbf{r} - \mathbf{r}') \cdot (\mathbf{r} - \mathbf{r}')} = \sqrt{r^2 + (r')^2 - 2\mathbf{r} \cdot \mathbf{r}'} = r\sqrt{1 + \frac{(r')^2}{r^2} - 2\frac{\hat{\mathbf{r}} \cdot \mathbf{r}'}{r}}, \quad (2)$$

expanding the contents under the square root with the small argument Taylor expansion

$$(1 + \varepsilon)^p \approx 1 + p\varepsilon + \mathcal{O}(\varepsilon^2) \quad (3)$$

we obtain

$$|\mathbf{r} - \mathbf{r}'| \approx r \left(1 - \frac{\hat{\mathbf{r}} \cdot \mathbf{r}'}{r} + \frac{1}{2} \frac{(r')^2}{r^2} \right) = r - \hat{\mathbf{r}} \cdot \mathbf{r}' + \mathcal{O}\left(\frac{1}{r}\right), \quad (4)$$

which yields

$$\frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r}-\mathbf{r}'|} \approx \frac{1}{r} e^{-jkr} e^{jk\hat{\mathbf{r}} \cdot \mathbf{r}'}. \quad (5)$$

Having a sum of isotropic antenna elements of this kind and neglecting mutual coupling, we can obtain our total field as

$$\sum_{n=1}^N \frac{1}{r} e^{-jkr} e^{jk\hat{\mathbf{r}} \cdot \mathbf{r}'_n} = \frac{1}{r} e^{-jkr} \sum_{n=1}^N e^{jk\hat{\mathbf{r}} \cdot \mathbf{r}'_n} = \frac{1}{r} e^{-jkr} AF, \quad (6)$$

where we have defined our array factor (AF) as the contribution from the sum of isotropic antenna elements. For cases where our antenna elements are not isotropic, we add on a factor E_e to obtain the total field

$$|F(\theta, \varphi)|^2 = |E_e(\theta, \varphi)AF(\theta, \varphi)|^2 = \left| E_e(\theta, \varphi) \sum_{n=1}^N e^{jk\hat{r} \cdot \mathbf{r}'_n} \right|^2. \quad (7)$$

By specifying our array geometry we can express the sum in 7 more easily. We define our array as a 2D uniform array consisting of N rows and M columns, placed in the xy-plane, with an element distance d :

$$\mathbf{r}'_{m,n} = x_m \hat{\mathbf{x}} + y_n \hat{\mathbf{y}}, \quad x_m = md, \quad y_n = nd. \quad (8)$$

Calculating the scalar product of 7, with the source-vector of 8, we have

$$AF = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{jk[x_m \sin \theta \cos \varphi + y_n \sin \theta \sin \varphi]}. \quad (9)$$

We also introduce an element unique phase shift $p_{m,n}$ where

$$p_{m,n} = e^{-jk[x_m \sin \theta_0 \cos \varphi_0 + y_n \sin \theta_0 \sin \varphi_0]}. \quad (10)$$

This yields us

$$AF = \sum_{m=0}^{M-1} p_m e^{jk[x_m \sin \theta \cos \varphi]} \sum_{n=0}^{N-1} p_n e^{jk[y_n \sin \theta \sin \varphi]}. \quad (11)$$

We begin by rewriting the inner sum of 11

$$\sum_{n=0}^{N-1} p_n e^{jk[y_n \sin \theta \sin \varphi]} = \sum_{n=0}^{N-1} e^{jk[nd \sin \theta \sin \varphi - nd \sin \theta_0 \sin \varphi_0]} = \sum_{n=0}^{N-1} \left(e^{jkd[\sin \theta \sin \varphi - \sin \theta_0 \sin \varphi_0]} \right)^n. \quad (12)$$

Here we can use the geometric series

$$\sum_{n=0}^{N-1} ax^n = a + ax + \dots + ax^{N-1} = a \frac{x^N - 1}{x - 1}, \quad (13)$$

which yields us

$$\begin{aligned} \sum_{n=0}^{N-1} \left(e^{jkd[\sin \theta \sin \varphi - \sin \theta_0 \sin \varphi_0]} \right)^n &= \sum_{n=0}^{N-1} \left(e^{j[\psi - \psi_0]} \right)^n = \frac{e^{jN[\psi - \psi_0]} - 1}{e^{j[\psi - \psi_0]} - 1} = \\ &\frac{e^{j\frac{N}{2}[\psi - \psi_0]} \left[e^{j\frac{N}{2}[\psi - \psi_0]} - e^{-j\frac{N}{2}[\psi - \psi_0]} \right]}{e^{j\frac{[\psi - \psi_0]}{2}} \left[e^{j\frac{[\psi - \psi_0]}{2}} - e^{-j\frac{[\psi - \psi_0]}{2}} \right]} = \frac{e^{j\frac{N}{2}[\psi - \psi_0]} \left[2j \sin \left(\frac{N}{2}[\psi - \psi_0] \right) \right]}{e^{j\frac{[\psi - \psi_0]}{2}} \left[2j \sin \left(\frac{1}{2}[\psi - \psi_0] \right) \right]} = \\ &e^{j[\psi - \psi_0](\frac{N-1}{2})} \frac{\sin \left(\frac{N}{2}[\psi - \psi_0] \right)}{\sin \left(\frac{1}{2}[\psi - \psi_0] \right)}, \end{aligned} \quad (14)$$

where $\psi = kd \sin \theta \sin \varphi$ and $\psi_0 = kd \sin \theta_0 \sin \varphi_0$. For the outer sum of 11, we perform the same operations as was done in 14. This yields us

$$AF = e^{j[\psi - \psi_0](\frac{N-1}{2})} e^{j[\zeta - \zeta_0](\frac{M-1}{2})} \frac{\sin \left(\frac{N}{2}[\psi - \psi_0] \right)}{\sin \left(\frac{1}{2}[\psi - \psi_0] \right)} \frac{\sin \left(\frac{M}{2}[\zeta - \zeta_0] \right)}{\sin \left(\frac{1}{2}[\zeta - \zeta_0] \right)}, \quad (15)$$

where $\zeta = kd \sin \theta \cos \varphi$ and $\zeta_0 = kd \sin \theta_0 \cos \varphi_0$. From the acquired expression, it is clear that the maximum array factor AF is in the direction $\hat{r}(\theta_0, \varphi_0)$. However, one can also see that if kd is large enough, there will be multiple maximums over the unit sphere.

We can further calculate the directivity D of the array factor

$$D(\theta, \varphi) = \frac{4\pi |AF(\theta, \varphi)|^2}{\int_{\Omega} |AF|^2 d\Omega}. \quad (16)$$

The integral in the denominator of 16 is given by

$$\int_0^{2\pi} \int_0^{\pi} \left| \frac{\sin \left(\frac{N}{2}[\psi - \psi_0] \right)}{\sin \left(\frac{1}{2}[\psi - \psi_0] \right)} \frac{\sin \left(\frac{M}{2}[\zeta - \zeta_0] \right)}{\sin \left(\frac{1}{2}[\zeta - \zeta_0] \right)} \right|^2 \sin \theta d\theta d\varphi. \quad (17)$$

2.2 Array geometries

2.2.1 Basic geometries

We start by analyzing a standard phased array geometry: the four-by-four uniform phased array (see Figure 1).

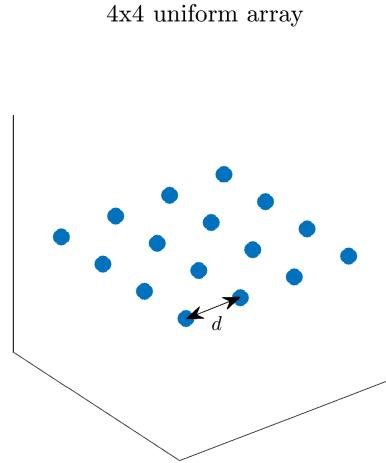


Figure 1: Uniform four-by-four array with distance d between the elements.

By using the geometry suggested in Figure 1 and operating on a frequency such that the uniform element distance d corresponds to $\lambda/2$, where λ is the wavelength, we get a far-field pattern calculated from the array factor AF , that can be seen in Figure 2 (a). Furthermore we can see the complete shape of the far-field in relation to the antenna elements in Figure 2 (b).

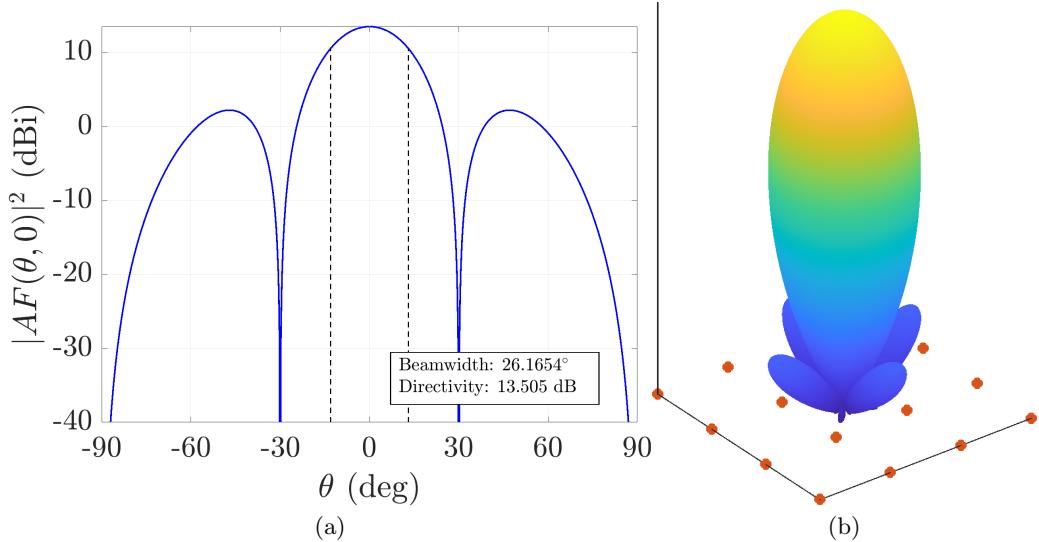


Figure 2: The far-field of the four-by-four array, with a uniform distance d corresponding to $\lambda/2$ in (a). The three-dimensional far-field pattern of the four-by-four array in (b).

By extending the uniform element distance d such that it now corresponds to λ , we get an interesting result, seen in Figure 3. We now get what is called grating lobes. These grating lobes are sidelobes that have the same gain as the mainbeam, although in another direction. However, the mainbeam beamwidth is smaller than for the array with a uniform element distance of $\lambda/2$.

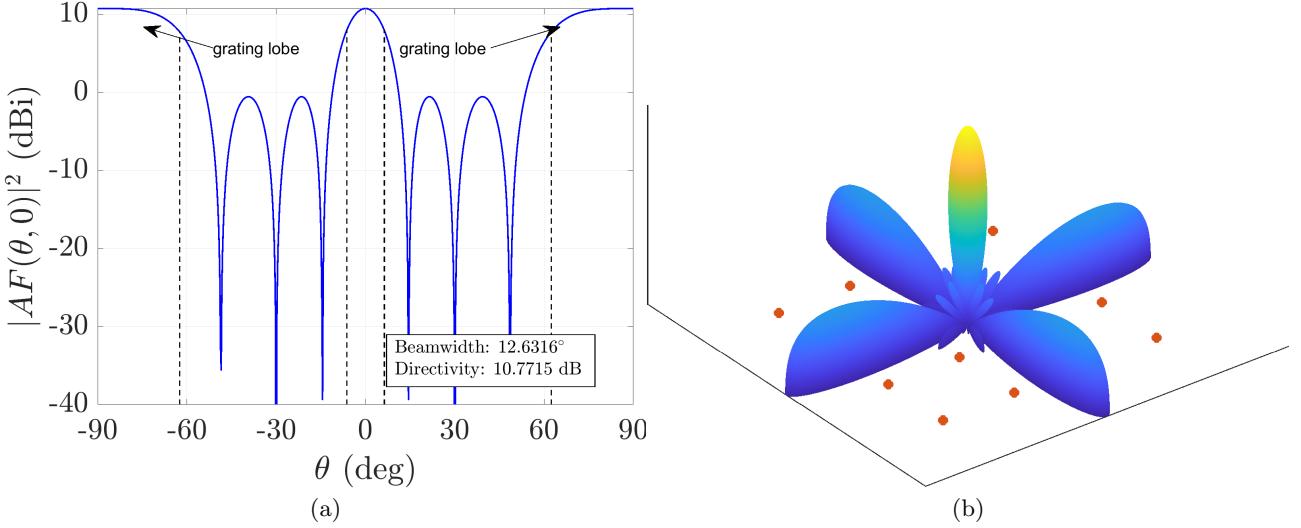


Figure 3: The far-field of the four-by-four array, with a uniform distance d corresponding to λ in (a). The three-dimensional far-field pattern of the four-by-four array in (b).

Then lastly, if we decrease the uniform element distance to $\lambda/4$ we get the result seen in Figure 4.

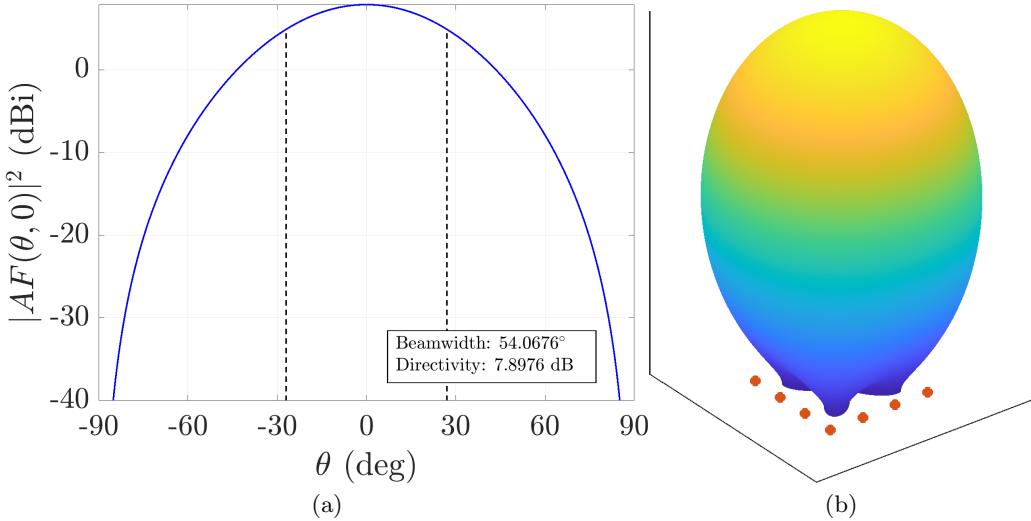


Figure 4: The far-field of the four-by-four array, with a uniform distance d corresponding to $\lambda/4$ in (a). The three-dimensional far-field pattern of the four-by-four array in (b).

2.2.2 Adaptive array configuration

As explained in the previous section, the element distance has a large impact on the far-field of the array. We saw that extending the distance between the elements can increase the beamwidth (and for some cases the directivity), but increase it too much and there will appear grating lobes. When working on small band arrays, one typically wants to have the distance between the elements to be around $\lambda/2$ if one wants to maximize the directivity and avoid grating lobes. For wideband arrays, however, there is a problem that one will run in to after a while.

For a wideband uniform 2D phased array system where the objective is to have the highest directivity possible, the first step would be to design the uniform element distance after the wavelength corresponding to the highest frequency of the intended band. This would ensure a very high directivity for the upper frequencies of the intended band. For the lower frequencies the element distance appears as a very small distance in terms of the lower frequencies wavelengths. This means the directivity for the lower frequencies gets lower than the directivity for the higher frequencies. It is however possible to expand the distance between the elements by not using them. By ignoring a well-chosen group of array elements, the distance between them can appear longer, however at the cost of using fewer elements. We can thus reconfigure our phased array system to maintain

a higher directivity at lower frequencies. To do this, we have defined a couple of "configuration modes" that determines which array element to disable (in this case for an eight-by-eight array, see Figure 5).

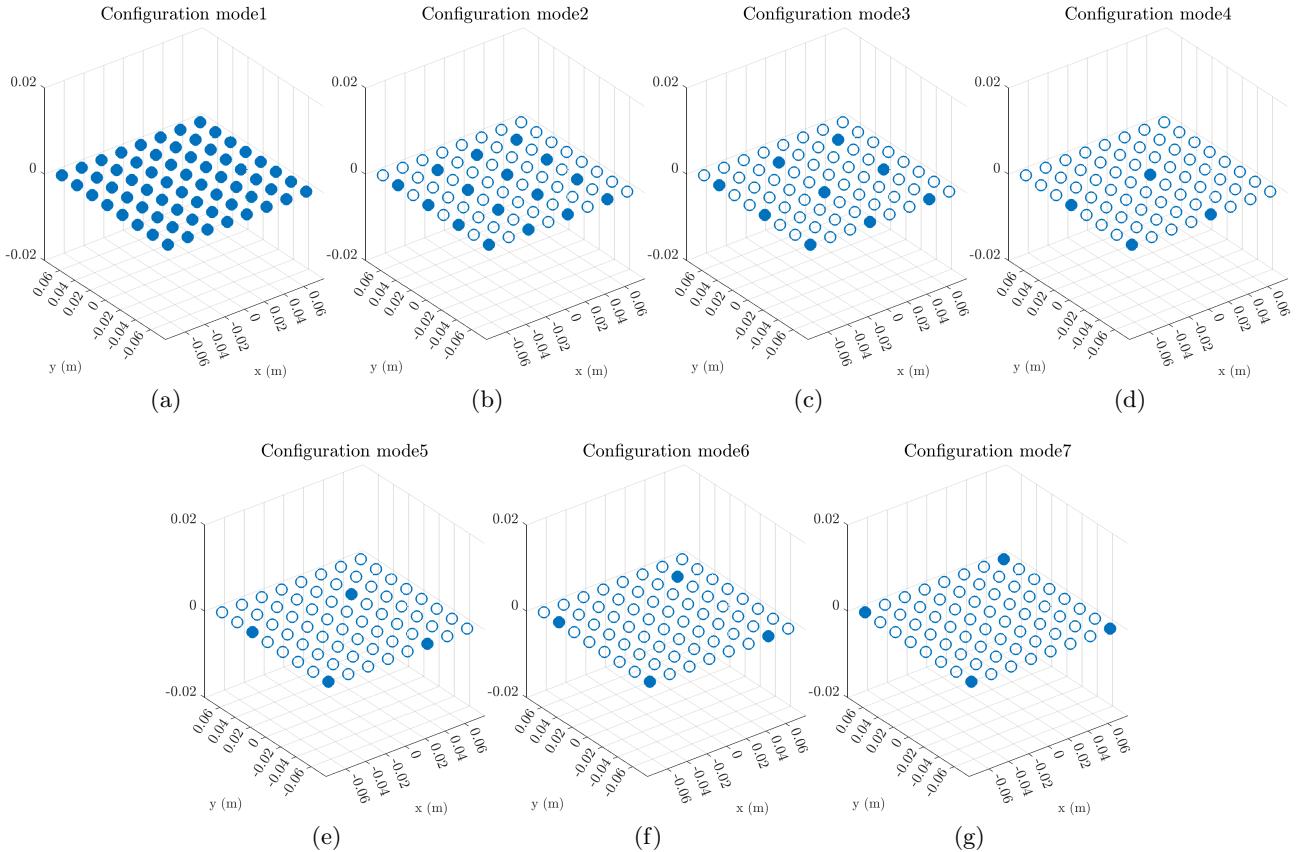


Figure 5: The different configuration modes for an eight-by-eight uniform array, where a solid circle indicates an array element in use and a circle with white face color indicates an array element not in use.

We can test these configuration modes out by creating an eight-by-eight uniform array with an element distance d of 20 mm. This corresponds roughly to $\lambda/2$ for the frequency 8.5 kHz (for a wave velocity of 343 m/s). We can then calculate the broadside directivity of the array at an operating frequency of 1.2 kHz. This yields us the result seen in Figure 6.

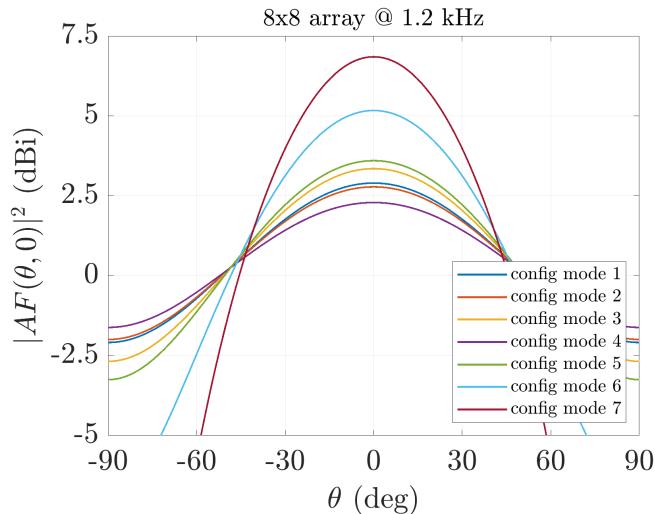


Figure 6: Uniform four-by-four array with distance d between the elements.

As can be seen in Figure 6, it is more beneficial to use the configuration mode 7 to obtain the highest directivity at 1.2 kHz. It is however important to notice that the drawback of this method is that the absolute

gain is lower when disabling array elements. If one is only interested in the magnitude of the signal, then it is always beneficial to using all of the array elements. If the objective however is to have the largest contrast of the incoming signal with dependence of the direction, then it is beneficial to use the configuration modes.

For the same array system used for Figure 6, we can test for multiple frequencies to see how much we can increase the directivity by using the configuration modes. Doing so, we must add one constraint. When disabling array elements to enlarge the element distance, we can not enlarge it to a point where the distance becomes larger than $\lambda/2$ for the intended frequency. The result for the proposed test can be seen in Figure 7

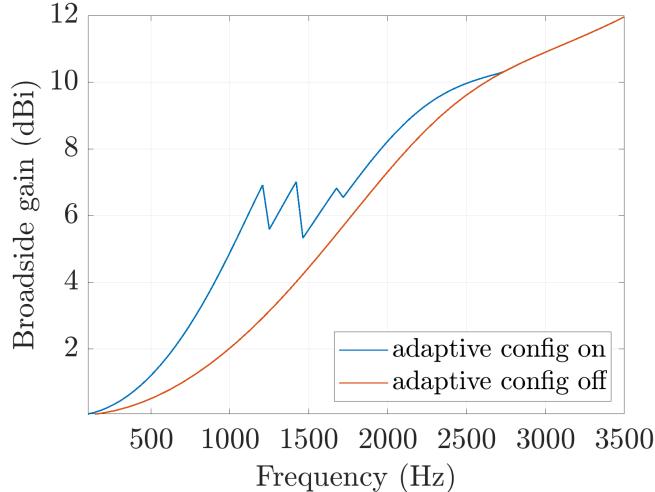


Figure 7: The broadside gain of the eight-by-eight uniform array for frequencies from 100 - 3500 Hz, with adaptive configuration and without adaptive configuration.

To know which configuration mode will give the highest directivity for a given frequency, one must try all the eligible configuration modes. This means that the directivity of the array system using one specific configuration mode must be calculated. To then integrate this solution to a system, one must either be able to calculate the integral given in 17 fast, or do all of the calculations beforehand. Furthermore, the suggested configuration modes in Figure 5 may not be the configuration modes that can yield the highest directivity for a given frequency. There can be a lot of patterns realized on the eight-by-eight grid, but one of the patterns will yield the highest directivity in a given direction for a given frequency.

2.2.3 Multiple arrays

We can use multiple arrays placed in a certain manner to obtain different kind of far-field patterns. The easiest way of placing these arrays would be the same way we placed our antenna elements; uniformly. Such a placement of arrays can be seen in Figure 8.

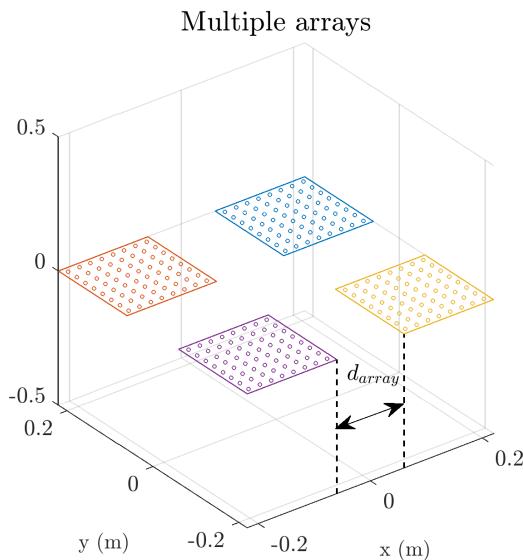


Figure 8: Multiple arrays with a uniform distance d_{array} from each other.

By adjusting this array distance d_{array} , we can get different far-field patterns. These far-field patterns can be seen in Figure 9.

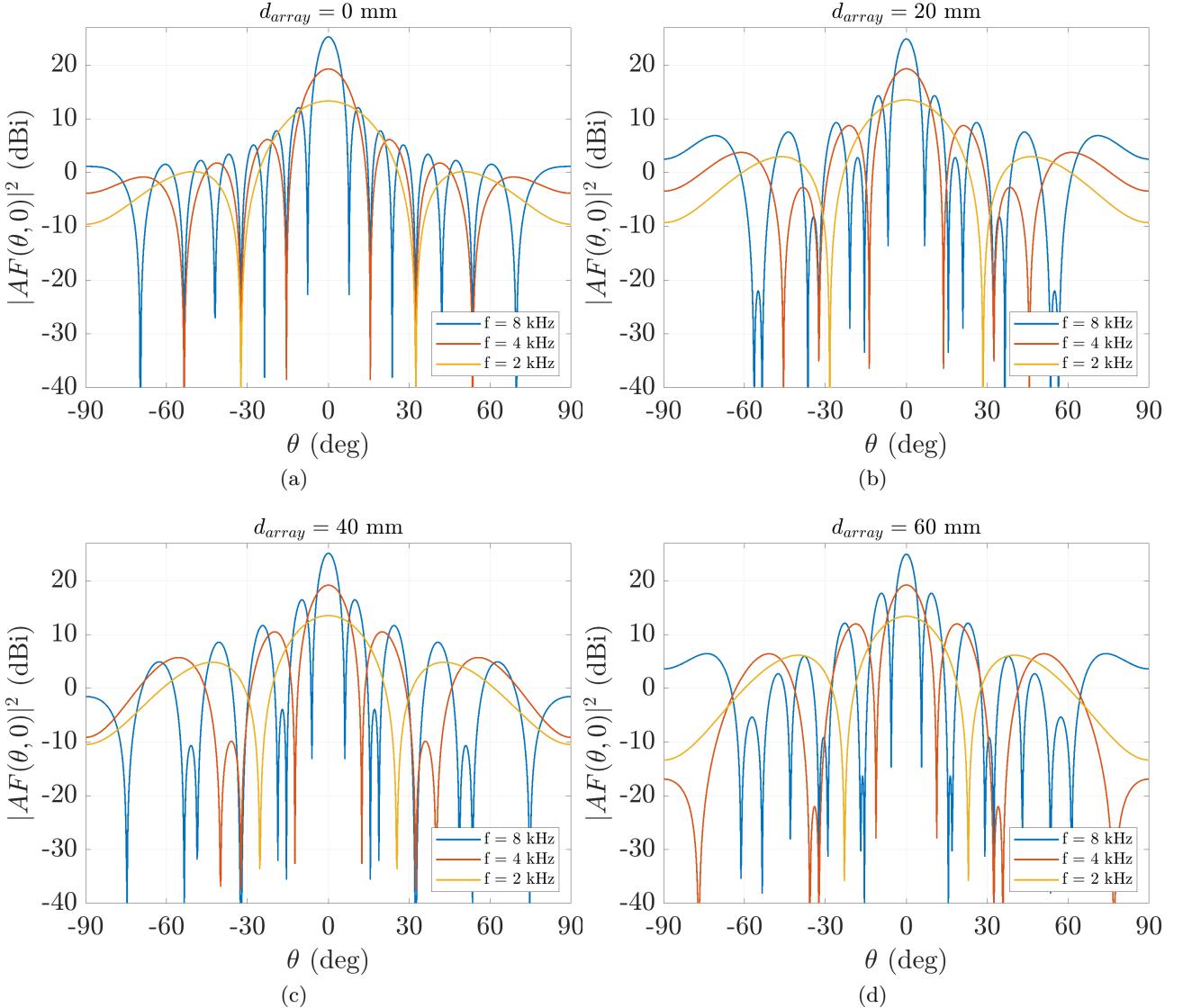


Figure 9: The different far-field patterns of a system consisting of four eight-by-eight arrays, with a uniform array distance of 0 mm in (a), 20 mm in (b), 40 mm in (c), and 60 mm in (d).

2.2.4 Multiple arrays with adaptive array configuration

Combining adaptive array configuration with multiple array antennas yields interesting results. We will however only use the simple configuration modes derived earlier. Also since adaptive array configuration yields distinguishable results at lower frequencies from not using adaptive array configuration, lower frequencies will be investigated more than higher frequencies. For four eight-by-eight arrays with a uniform array distance d_{array} of 60 mm, we get the configuration modes as seen in Figure 10.

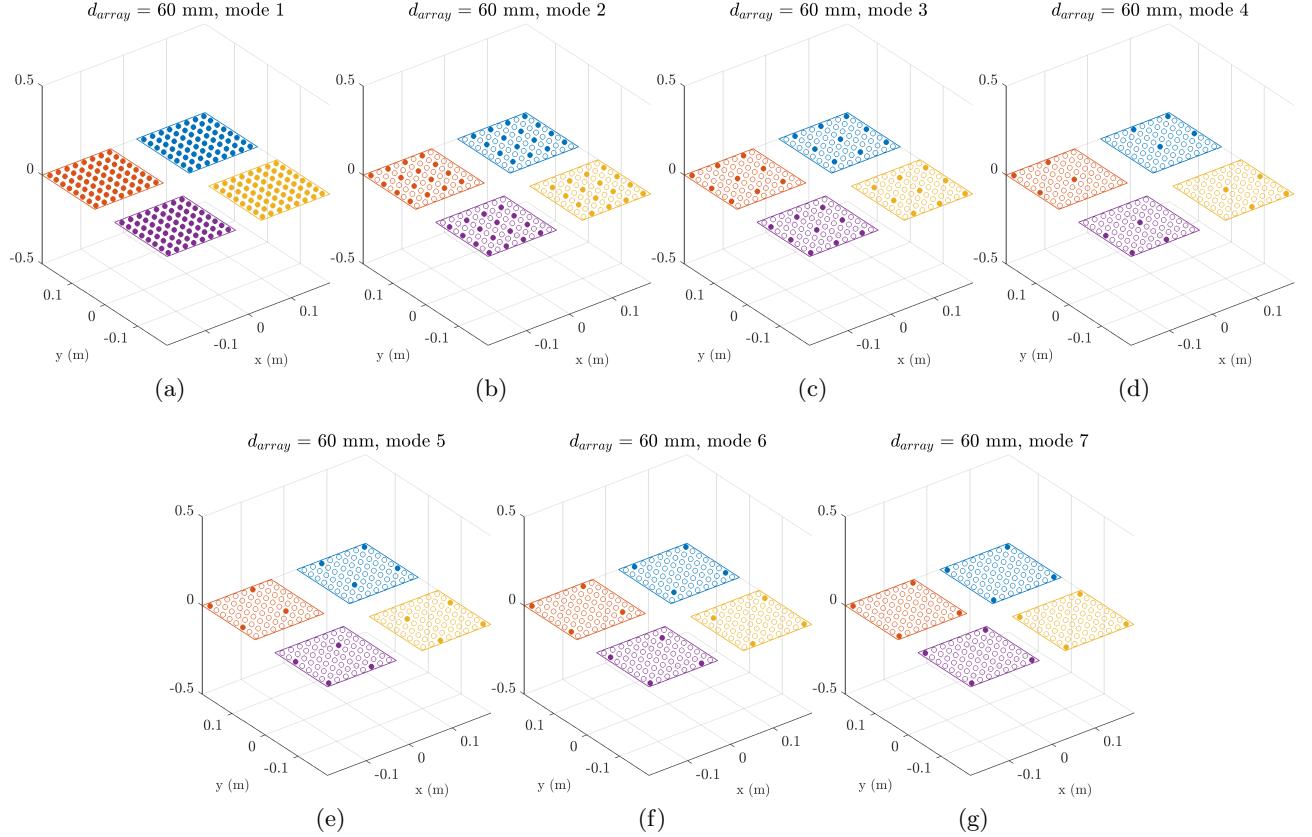


Figure 10: The different configuration modes for four eight-by-eight uniform array with an array distance d_{array} of 60 mm, where a solid circle indicates an array element in use and a circle with white face color indicates an array element not in use.

Operating the four eight-by-eight arrays at 2000 Hz yields the far-field pattern seen in Figure 11.

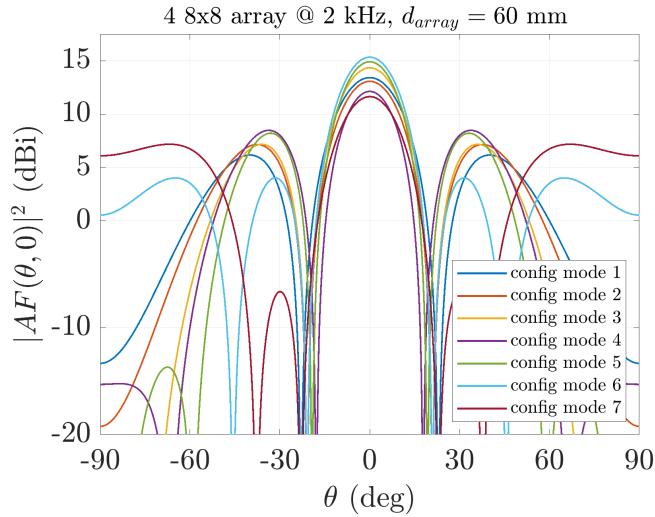


Figure 11: The far field of the four eight-by-eight arrays when using different configuration modes.

3 The beam forming algorithm

3.1 Theory

The proposed beam forming algorithm is based upon the far-field approximation of spherical waves. This approximation will coincide well if we have the following conditions met: $kr \gg 1$ and $r \gg r'$. We can thus approximate the spherical wave as done in Section 2.1

$$\frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r}-\mathbf{r}'|} \approx \frac{1}{r} e^{-jkr} e^{jkr \hat{\mathbf{r}} \cdot \mathbf{r}'} . \quad (18)$$

For larger distances we can see, from (5), that the signals generated on each antenna element has the same amplitude as well as a phase-shift only dependent on the direction of the incoming signal, the location of the antenna element, and the angular frequency of the wave. This means that we can add element-unique phase-shifts to the generated signals on each element to obtain a constructive interference in a desired direction. In other words, we can listen in a certain direction. For now, however, we can only listen for narrowband signals (single frequencies). To obtain a wide-band solution, we must filter our incoming signals into multiple small bands, and then apply our frequency dependent phase-shifts.

From (5) we can also see that if we want constructive interference in the $\hat{\mathbf{r}}$ direction, we need to multiply with a phasor p such that the element-unique signals are in phase for every frequency band. One solution is

$$p = e^{-jkr \hat{\mathbf{r}} \cdot \mathbf{r}'_i} . \quad (19)$$

In time domain, this phase shift is simply $-k\hat{\mathbf{r}} \cdot \mathbf{r}'_i$. Performing this element-unique phase-shift would grant constructive interference in the desired direction.

To perform a phase-shift in the time domain, we need some kind of filter. Since we are working with discrete signals, we will need a digital filter. To create such a filter, we first start with a trigonometric relation

$$\sin(\alpha + \beta) = \sin \alpha \cos \beta + \cos \alpha \sin \beta . \quad (20)$$

Let $\alpha = 2\pi\nu n$ and $\beta = \phi_0$ and we instead have

$$\sin(2\pi\nu n + \phi_0) = \sin(2\pi\nu n) \cos \phi_0 + \cos(2\pi\nu n) \sin \phi_0 . \quad (21)$$

We then rewrite $\cos(2\pi\nu n)$ as $1/(2\pi\nu) \frac{d}{dn} \sin(2\pi\nu n)$

$$\sin(2\pi\nu n + \phi_0) = \sin(2\pi\nu n) \cos \phi_0 + \frac{\sin \phi_0}{2\pi\nu} \frac{d}{dn} \sin(2\pi\nu n) . \quad (22)$$

We can approximate the derivative in (22) with numerical differentiation

$$\sin(2\pi\nu n + \phi_0) = \sin(2\pi\nu n) \cos \phi_0 + \frac{\sin \phi_0}{2\pi\nu} \frac{1}{2} [\sin(2\pi\nu(n+1)) - \sin(2\pi\nu(n-1))] . \quad (23)$$

If we thus have a discrete sine signal $x[n] = \sin(2\pi\nu n)$ and we want to phase-shift it ϕ_0 , we can thus utilize the derived difference equation

$$y[n] = x[n] \cos \phi_0 + \frac{\sin \phi_0}{4\pi\nu} (x[n+1] - x[n-1]) . \quad (24)$$

From here, we would like to further specify the phase shift ϕ_0 to be applied in order to get constructive interference in a desired direction. We do this by calculating the dot product in 19. For this beamforming algorithm, the array (or arrays) are assumed to lie in the xy-plane (although changing this is not a problem). We can thus express our element vector \mathbf{r}'_i as

$$\mathbf{r}'_i = x_i \hat{\mathbf{x}} + y_i \hat{\mathbf{y}} . \quad (25)$$

Its worth noticing that we use the same index i on both the x coordinate as well as the y coordinate. This is because in this algorithm, we stack our element vectors into a matrix \mathbf{r}' . This means that we can still have a 2D array, although we only have one index i .

$$\mathbf{r}' = \begin{bmatrix} x_1 & x_2 & \dots & x_M \\ y_1 & y_2 & \dots & y_M \\ 0 & 0 & \dots & 0 \end{bmatrix} , \quad (26)$$

where M indicates the last element in the array.

We now only need to perform the dot product between 25 and $\hat{\mathbf{r}}$

$$\mathbf{r}'_i \cdot \hat{\mathbf{r}} = (x_i \hat{\mathbf{x}} + y_i \hat{\mathbf{y}}) \cdot (\sin \theta \cos \varphi \hat{\mathbf{x}} + \sin \theta \sin \varphi \hat{\mathbf{y}} + \cos \theta \hat{\mathbf{z}}) = x_i \sin \theta \cos \varphi + y_i \sin \theta \sin \varphi. \quad (27)$$

The frequency dependent phase shift to apply thus becomes

$$\phi_i = -k(x_i \sin \theta \cos \varphi + y_i \sin \theta \sin \varphi). \quad (28)$$

We have thus derived an element-unique phase shift (28), and the means to apply that phase shift (24).

3.2 Beamforming of recorded data

3.2.1 Basic idea

When performing the beamforming algorithm, there are mainly two ways of operating; scanning the whole space to determine in what direction there could be a source, or to simply perform the beamforming algorithm in one direction to enhance the signal coming from that direction. In this section, we will explain how the algorithm works in the simplest case; with recorded data.

We start with the signals from the 2D array

$$\begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^M \\ \dots & \dots & \dots & \dots \\ x_{n-2}^1 & x_{n-2}^2 & \dots & x_{n-2}^M \\ x_{n-1}^1 & x_{n-1}^2 & \dots & x_{n-1}^M \\ x_n^1 & x_n^2 & \dots & x_n^M \end{bmatrix}, \quad (29)$$

where x_n^i indicates the n:th sample recorded from element i . Every column of 29 thus refer to one discrete signal of one element. We can rewrite our data matrix into a row vector of column vectors (a matrix)

$$[x^1 \ x^2 \ \dots \ x^M]. \quad (30)$$

In 30, every element (x^1 , x^2 etc.) represents the complete signal recorded from one element. From here, we want to filter every element signal into narrow band signals. If we want to filter our signals into 20 narrowband signals, we need to filter every element signal 20 times. A block diagram can be seen in Figure 12.

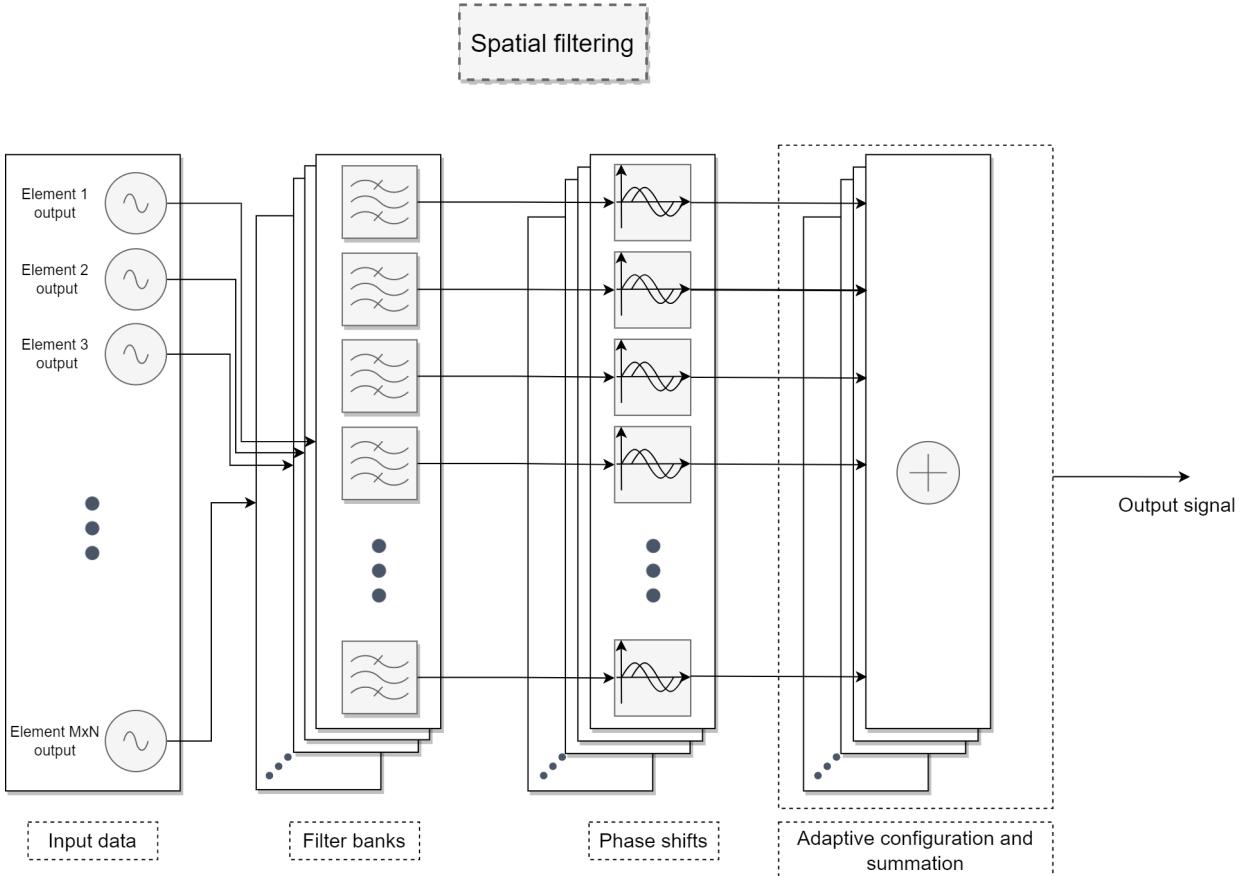


Figure 12: Block diagram of beamforming algorithm acting on recorded data. In this example, the data is filtered in an arbitrary amount of bands. The output signal is the spatially filtered signal.

The algorithm in Figure 12 will only spatially filter the signal in one direction. This means that the output $\mathbf{y}(\theta, \varphi)$ will be dependent on how the phase filters are configured. This algorithm has to run for multiple directions in order to scan the whole space.

3.2.2 Filter banks

Since we need to filter our signals into many small band signals, we will use a filter bank. What this means is that we simply will apply many narrow band-pass filters. We will also only use FIR-filters to avoid resonance (finite impulse response).

The easiest and most convenient way of designing the filter bank is to use the fir1 filter (see matlab's description) and to space out the cut-off frequencies linearly in the desired frequency band.

Applying these digital FIR-filters can be done in time domain

$$y[n] = \frac{1}{a_0} \sum_{i=0}^P b_i x[n-i], \quad (31)$$

where $y[n]$ is the output signal, $x[n]$ is the input signal, a_0 is the magnitude of the output signal, b_i is the filter coefficients, and P is the filter order. The filter coefficients can be obtained using Matlab (see fir1 function). When using the fir1 function in Matlab, one has to define the filter order, and the two cut-off frequencies (we want a band-pass filter). What is recommended to do is to define a center frequency for the band-pass filter, and then to define the cut-off frequencies as $\pm 1\%$ of the center frequency.

As can be seen in Equation 31, the output signal is a scalar value, and not a vector. We thus perform the arithmetic on the right hand side of 31 for every n , where the amount of operations depends on how many samples long our recorded data is. We can also see that there will be problematic to output the first filtered output $y[1]$. When we filter the sampled signal $x[n]$, we rely on older samples. In the very beginning, there may not exist any older sample/samples. We thus has to make the best out of the situation, and use the oldest samples we can get our hands on.

3.2.3 Phase shift filters

The phase shift filters can be realized straight from the theory in 24 and 28. To perform phase shift filtering as the block diagram in Figure 12 suggests, one must however create a phase shift function that can take in the two parameters k and i . These two parameters refer to the frequency of which the signal has been filtered to (k), and which antenna element the signal is coming from (i). Furthermore, the phase filter function must know the direction (θ, φ) to spatially filter the incoming signal.

The phase shifting filter will also need to know the normalized frequency ν , which means that the sampling frequency f_s must be known.

3.2.4 Adaptive configuration

Adaptive configuration is a weighting-technique that enhances the beamforming algorithms precision at lower frequencies. It works by not adding the signal from some antenna elements when the signal has been filtered with a band-pass filter with low cut-off frequencies. How to choose which antenna element to ignore and when is at this stage decided by a simple look-up table. First, if the wavelength of the frequency is too long compared to the distance between the antenna elements, a certain configuration mode will be activated. This configuration mode will tell which element to stay on, and which to be turned off.

These configuration modes are heavily dependent on the geometry of the array antenna/antennas used. For further reading, see section 2.2.2.

3.2.5 Multiple directions

If we want to perform the beamforming algorithm in Figure 12 for multiple angles to sweep a surface, there are some things to take in mind.

Usually, we want to sweep over a surface in a Cartesian coordinate system. We then have to make some sort of translation when we use the beamforming algorithm, since it only takes in the angle-parameters θ and φ , see Figure 13.

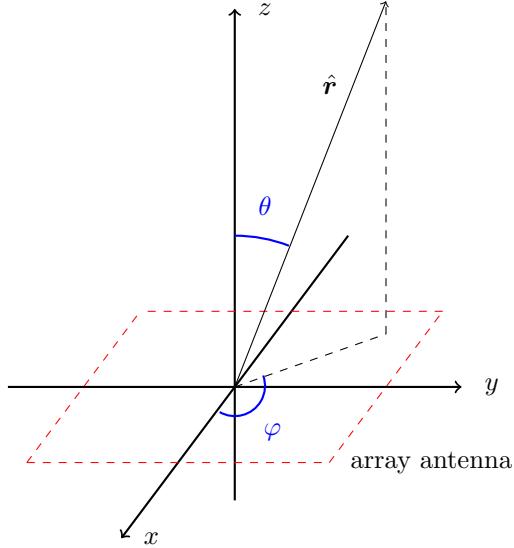


Figure 13: The beamforming listening direction $\hat{r}(\theta, \varphi)$ and its relation to the Cartesian coordinate system.

What we mathematically do here is to project a rectangular cut-out of a sphere onto the xy-plane, see Figure 14.

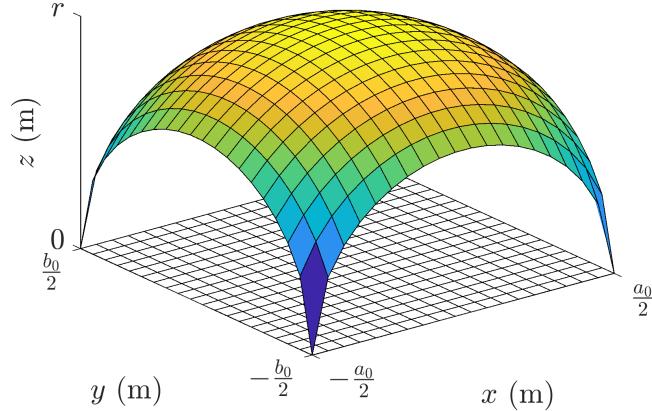


Figure 14: The rectangular cut-out of a sphere projected onto the xy-plane.

We define the rectangle as

$$-\frac{a_0}{2} \leq x \leq \frac{a_0}{2}, \quad -\frac{b_0}{2} \leq y \leq \frac{b_0}{2}. \quad (32)$$

We further need to define a radius to the sphere we cut from. This radius will have some constraints

$$\frac{1}{2} \sqrt{a_0^2 + b_0^2} \leq r. \quad (33)$$

We now have everything we need to translate the Cartesian coordinates to spherical coordinates

$$\theta = \arccos \left(\frac{\sqrt{r^2 - x^2 - y^2}}{r} \right), \quad \varphi = \arctan \frac{y}{x}. \quad (34)$$

The translation for φ will need some modification when used in a function since the arctan-function will only yield values in the region $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. Instead when implementing a function, use `atan2(y, x)`.

With the proposed translation, it is possible to use the beamforming algorithm for every coordinate in a grid. Doing so yields a data matrix consisting of many output signals \mathbf{y} for every coordinate in this grid

$$\begin{bmatrix} \mathbf{y}_{1,1} & \mathbf{y}_{1,2} & \dots & \mathbf{y}_{1,B} \\ \mathbf{y}_{2,1} & \mathbf{y}_{2,2} & \dots & \mathbf{y}_{2,B} \\ \dots & \dots & \dots & \dots \\ \mathbf{y}_{A,1} & \mathbf{y}_{A,2} & \dots & \mathbf{y}_{A,B} \end{bmatrix}, \quad (35)$$

where A is the amount of rows in the grid, and B is the amount of columns in the grid. If we for every entry in matrix 35 take the dot product of the entry and itself (see Hadamard product, $\mathbf{y}_{1,1} \cdot \mathbf{y}_{1,1}$ etc.) we get a new matrix, where every entry would be proportional to the intensity in the direction represented by the entry

$$\mathbf{I}^{bf} = \begin{bmatrix} I_{1,1} & I_{1,2} & \dots & I_{1,B} \\ I_{2,1} & I_{2,2} & \dots & I_{2,B} \\ \dots & \dots & \dots & \dots \\ I_{A,1} & I_{A,2} & \dots & I_{A,B} \end{bmatrix}. \quad (36)$$

This intensity map \mathbf{I}^{bf} is something we can plot to get perception of where in the space there can be sources.

3.2.6 Post processing

It is possible to enhance the intensity map \mathbf{I}^{bf} such that the location of the sources is much more clear. To do this, We define the following matrices

$$C_{a,b}^x = (I_{a,(b+1)}^{bf} - I_{a,(b-1)}^{bf})/x_{step}, \quad 1 \leq a \leq A, \quad 1 < b < B, \quad (37)$$

$$C_{a,b}^y = (I_{(a+1),b}^{bf} - I_{(a-1),b}^{bf})/y_{step}, \quad 1 < a < A, \quad 1 \leq b \leq B, \quad (38)$$

$$D_{a,b} = 1/|C_{a,b}^x + C_{a,b}^y|, \quad 1 < a < A, \quad 1 < b < B, \quad (39)$$

$$E_{a,b} = - \left[(C_{(a+1),b}^y - C_{(a-1),b}^y)/y_{step} + (C_{a,(b+1)}^x - C_{a,(b-1)}^x)/x_{step} \right], \quad 2 < a < A-1, \quad 2 < b < B-1, \quad (40)$$

where a and b are row indices and column indices respectively, x_{step} is the distance between the points in the grid in the x-direction, and y_{step} is the distance between the points in the grid in the y-direction. Now to enhance the intensity map \mathbf{I}^{bf} we perform the following operation

$$F_{a,b} = I_{a,b}^{bf} D_{a,b} E_{a,b}, \quad (41)$$

where \mathbf{F} is the enhanced intensity map.

3.2.7 Optimization

One part of the total spatial filtering algorithm that easily can be optimized is the two filtering stages that occur in series, see Figure 12. These two filtering stages can be expressed as one single filter, as Figure 15 suggests.

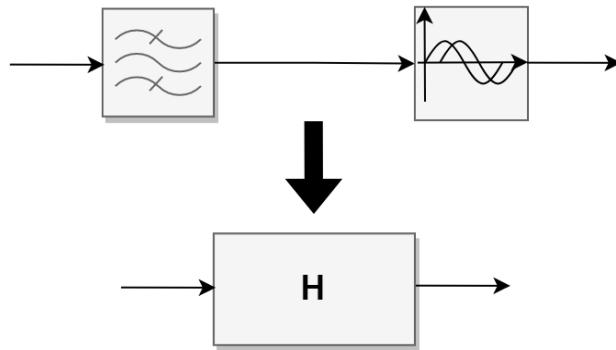


Figure 15: Treating the two filters in series as one complete filter \mathbf{H} .

We thus get the following FIR-filters in series

$$y[n] = \frac{1}{a_0} \sum_{i=0}^P b_i x[n-i], \quad z[n] = y[n] \cos \phi_0 + \frac{\sin \phi_0}{4\pi\nu} (y[n+1] - y[n-1]). \quad (42)$$

Inserting the first equation of 42 into the second equation of 42, we get

$$z[n] = \frac{\cos \phi_0}{a_0} \sum_{i=0}^P b_i x[n-i] + \frac{\sin \phi_0}{4\pi\nu a_0} \left(\sum_{i=0}^P b_i x[n-i+1] - \sum_{i=0}^P b_i x[n-i-1] \right). \quad (43)$$

From here we want to have the expression in Equation 43 such that the filter coefficients can be more easily obtained. To do this we perform some index manipulation

$$-i + 1 = -i' \implies \sum_{i=0}^P b_i x[n - i + 1] = \sum_{i'=-1}^{P-1} b_{i'+1} x[n - i'], \quad (44)$$

$$-i - 1 = -i' \implies \sum_{i=0}^P b_i x[n - i - 1] = \sum_{i'=1}^{P+1} b_{i'-1} x[n - i']. \quad (45)$$

performing the suggested index manipulation on Equation 43, and dropping the prime off of i , we get

$$z[n] = \frac{\cos \phi_0}{a_0} \sum_{i=0}^P b_i x[n - i] + \frac{\sin \phi_0}{4\pi\nu a_0} \left(\sum_{i=-1}^{P-1} b_{i+1} x[n - i] - \sum_{i=1}^{P+1} b_{i-1} x[n - i] \right), \quad (46)$$

which yields

$$\begin{aligned} z[n] = & \frac{\cos \phi_0}{a_0} \sum_{i=0}^P b_i x[n - i] + \frac{\sin \phi_0}{4\pi\nu a_0} \left(\sum_{i=1}^{P-1} b_{i+1} x[n - i] - \sum_{i=1}^{P-1} b_{i-1} x[n - i] \right) + \\ & \frac{\cos \phi_0}{a_0} (b_0 x[n] + b_P x[n - P]) + \frac{\sin \phi_0}{4\pi\nu a_0} (b_0 x[n + 1] + b_1 x[n] - b_{P-1} x[n - P] - b_P x[n - P - 1]). \end{aligned} \quad (47)$$

The expression in 47 is dependent on $n + 1$, which is usually not practical. Instead, we want to shift the whole expression. We do this with the following index manipulation

$$n + 1 = n'. \quad (48)$$

Dropping the prime, we get

$$\begin{aligned} z[n - 1] = & \frac{\sin \phi_0}{4\pi\nu a_0} b_0 x[n] + \left(\frac{\cos \phi_0}{a_0} b_0 + \frac{\sin \phi_0}{4\pi\nu a_0} b_1 \right) x[n - 1] + \sum_{i=2}^P \left(\frac{\cos \phi_0}{a_0} b_{i-1} + \frac{\sin \phi_0}{4\pi\nu a_0} (b_i - b_{i-2}) \right) x[n - i] \\ & \left(\frac{\cos \phi_0}{a_0} b_P - \frac{\sin \phi_0}{4\pi\nu a_0} b_{P-1} \right) x[n - P - 1] - \frac{\sin \phi_0}{4\pi\nu a_0} b_P x[n - P - 2] \end{aligned} \quad (49)$$

Since our input $x[n]$ is not dependent on our output $z[n]$, we can phase-shift the left hand side of 49 without any consequences. We can thus identify our new filter coefficients \mathbf{c} for the completely merged filter suggested in Figure 15

$$\begin{aligned} c_0 &= \frac{\cos \phi_0}{a_0}, \quad c_1 = \left(\frac{\cos \phi_0}{a_0} b_0 + \frac{\sin \phi_0}{4\pi\nu a_0} b_1 \right), \quad c_i = \left(\frac{\cos \phi_0}{a_0} b_{i-1} + \frac{\sin \phi_0}{4\pi\nu a_0} (b_i - b_{i-2}) \right), \quad 2 \leq i \leq P, \\ c_{P+1} &= \left(\frac{\cos \phi_0}{a_0} b_P - \frac{\sin \phi_0}{4\pi\nu a_0} b_{P-1} \right), \quad c_{P+2} = -\frac{\sin \phi_0}{4\pi\nu a_0} b_P. \end{aligned} \quad (50)$$

4 Emulated data

4.1 The geometry of the environment

In order to evaluate the performance of the beamforming algorithm, we must first generate an environment that has all the necessary components for the beamforming algorithm to work. This means that we must create an environment with an existing array antenna and sources for the array antenna to sense. Such an environment can be seen in Figure 16.

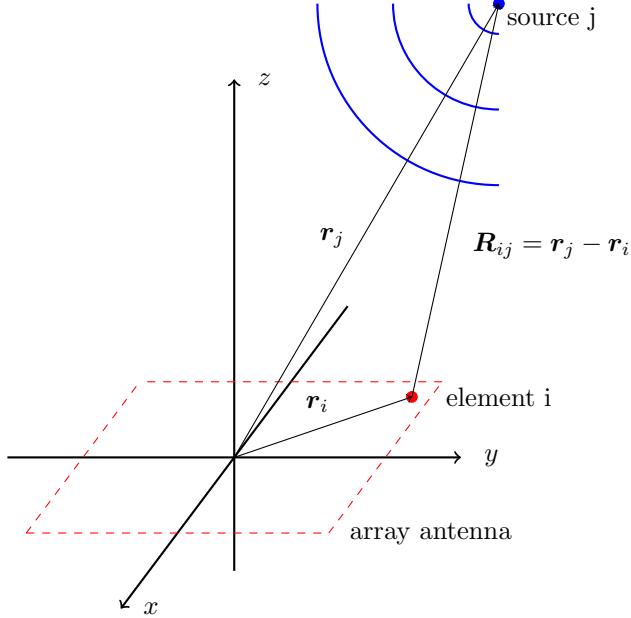


Figure 16: Location of the j:th source relative to the array antenna and its i:th element.

Every source's location is defined by two angles (θ and φ according to a spherical coordinate system) and a distance ρ from the origin. The emitted signal from the sources are defined by a discrete sum of sine waves with frequencies defined by the user. For example, the user could define a source's signal as a sum of two sine waves, one at 440 Hz, and the other at 659 Hz. The wave velocity of the emitted spherical waves can also be determined by the user, however, no relativistic effects are accounted for.

4.2 Generating signals

On each antenna element of the array antenna, there will be a signal generated. The signal generated will be dependent on how many sources there are, the frequencies of the sine waves emitted from the sources, and the distance between the antenna element and the sources. The distance $|\mathbf{R}_{ij}|$ between the i:th antenna element and the j:th source will have an effect on the amplitude and the phase of the generated signal on the i:th antenna element. The propagation of the waves generated by the sources are expected to behave as spherical waves

$$\frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}, \quad (51)$$

where the wave vector k is defined as $k = \omega/c$, where c is the wave velocity and ω is the angular frequency of the wave.

Each source will generate a signal that from its perspective would appear as

$$y_j(t) = \sum_{g=1}^G \sin(2\pi f_g t), \quad (52)$$

where the index g relates to the frequencies of the sine waves the complete signal consists of. We can then use this model together with the spherical wave 51 to calculate the generated signals on each antenna element

$$x_i(t) = \sum_{j=1}^J \sum_{g=1}^G \frac{1}{|\mathbf{r}_j - \mathbf{r}_i|} \sin(2\pi f_{jg} t - k_g |\mathbf{r}_j - \mathbf{r}_i|). \quad (53)$$

There is a slight modification that has to be done to this model to work with the proposed beamforming algorithm. The signal has to be a discrete signal rather than a time-continuous one. We can easily create a discrete signal by sampling the time-continuous signal

$$x_i[n] = \sum_{j=1}^J \sum_{g=1}^G \frac{1}{|\mathbf{r}_j - \mathbf{r}_i|} \sin(2\pi f_{jg} t[n] - k_g |\mathbf{r}_j - \mathbf{r}_i|), \quad (54)$$

where $t[n]$ is the discrete time, determined by a sampling frequency f_s .

In Figure 17 we can see two cases of when signals are generated with the model of 54.

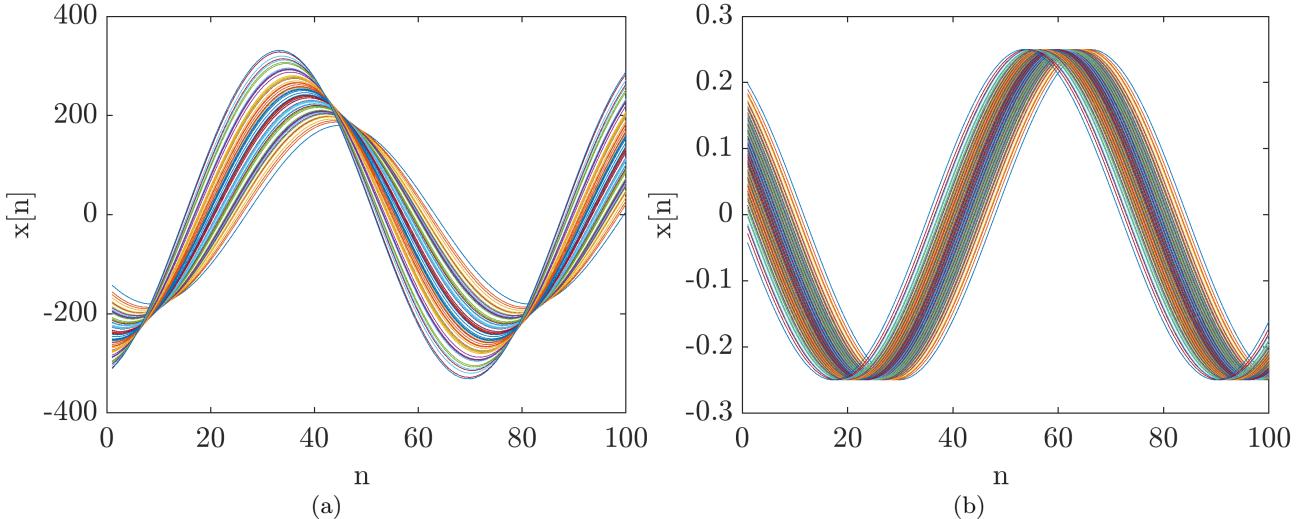


Figure 17: Generated signals on an eight by eight uniform array located in the origin and the xy-plane, where the source is at a distance of 0.2 m from the origin and at an angle of $\theta = \frac{\pi}{4}$ and $\varphi = \frac{\pi}{6}$, in (a). The generated signals on the same array as in (a), but with the source located 200 m from the origin and at an angle of $\theta = \frac{\pi}{4}$ and $\varphi = \frac{\pi}{6}$ in (b).

4.3 Beam forming algorithm performance on emulated data

We test the algorithm with the following settings presented in table 1, 2 and 3. This yields the result seen in Figure 18.

Table 1: Source settings

| Source | f_{start} (Hz) | f_{end} (Hz) | $N_{\text{sine waves}}$ | θ (deg) | φ (deg) | ρ (m) | t_{start} (s) | t_{end} (s) |
|--------|------------------|----------------|-------------------------|----------------|-----------------|------------|-----------------|---------------|
| 1 | 2600 | 3300 | 40 | 25 | 180 | 5 | 0 | 0.35 |
| 2 | 2700 | 2900 | 40 | 20 | 0 | 5 | 0.15 | 0.5 |

Table 2: Array settings

| Array | \mathbf{r}_a | columns | rows | uniform distance (m) |
|-------|----------------|---------|------|----------------------|
| 1 | (0,0) | 8 | 8 | 0.02 |

Table 3: Beam forming settings

| f_s (Hz) | t_{start} (s) | t_{end} (s) | a_0 (m) | b_0 (m) | r (m) | x_{step} (m) | y_{step} (m) | adaptive config. |
|------------|-----------------|---------------|-----------|-----------|------------|----------------|----------------|------------------|
| 16000 | 0 | 0.5 | 2 | 2 | $\sqrt{2}$ | 0.067 | 0.067 | ON |

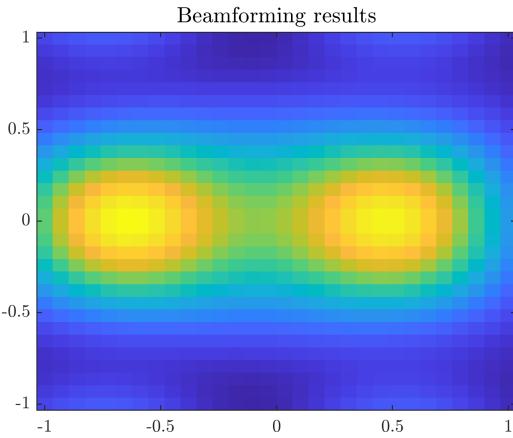


Figure 18: The normalized intensity map generated by the beamforming algorithm with the settings in Table 3.

The normalized enhanced intensity map and the correct locations of the sources can be seen in Figure 19.

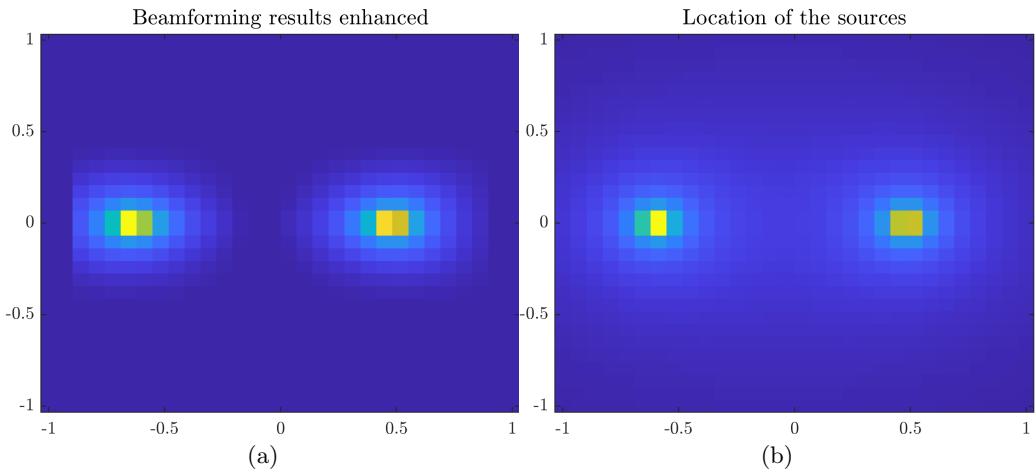


Figure 19: The enhanced intensity map in (a), and the correct location of the sources in (b).

Letting an island-finder function operate on the enhanced intensity map yields two islands with corresponding angles $\theta_1 = 26^\circ$, $\varphi_1 = 180^\circ$, $\theta_2 = 20^\circ$ and $\varphi_2 = 0^\circ$.

Listening in the direction of source 2 we get the result seen in Figure 20 (b).

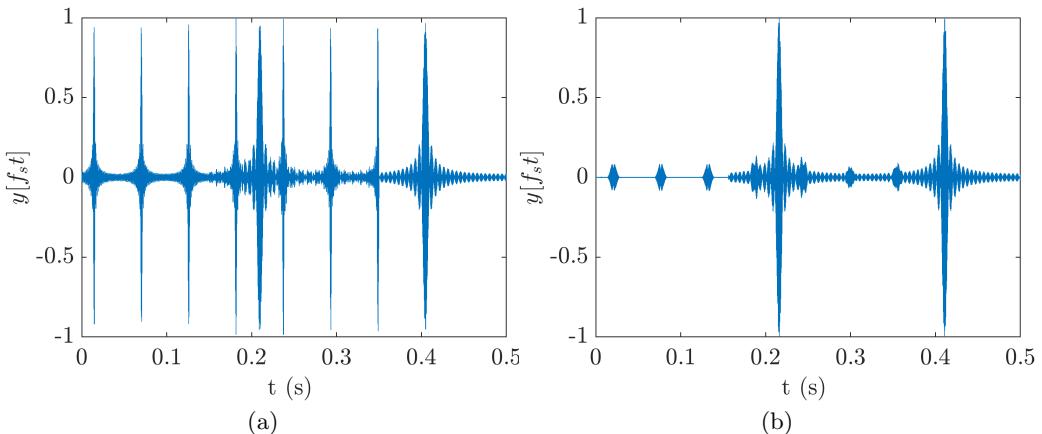


Figure 20: The raw signal received by the first antenna element in (a). The signal processed by the single direction beamforming algorithm in (b).

4.3.1 One source, one array

In this section, the beam forming algorithm will be evaluated depending on the frequency of the emitted sound signal. Only one array will be active, and the signals will be sampled at a frequency of $f_{sampling} = 15\,625$ Hz. One source have been placed 20 m away from the array, at an angle of $\theta = 30^\circ$ and $\phi = 20^\circ$. The real location of the source is illustrated in figure 21. The results at different frequencies can be found in figure 22.

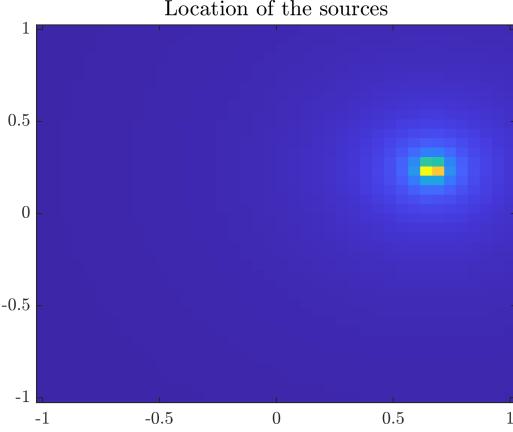


Figure 21: Actual location of one source.

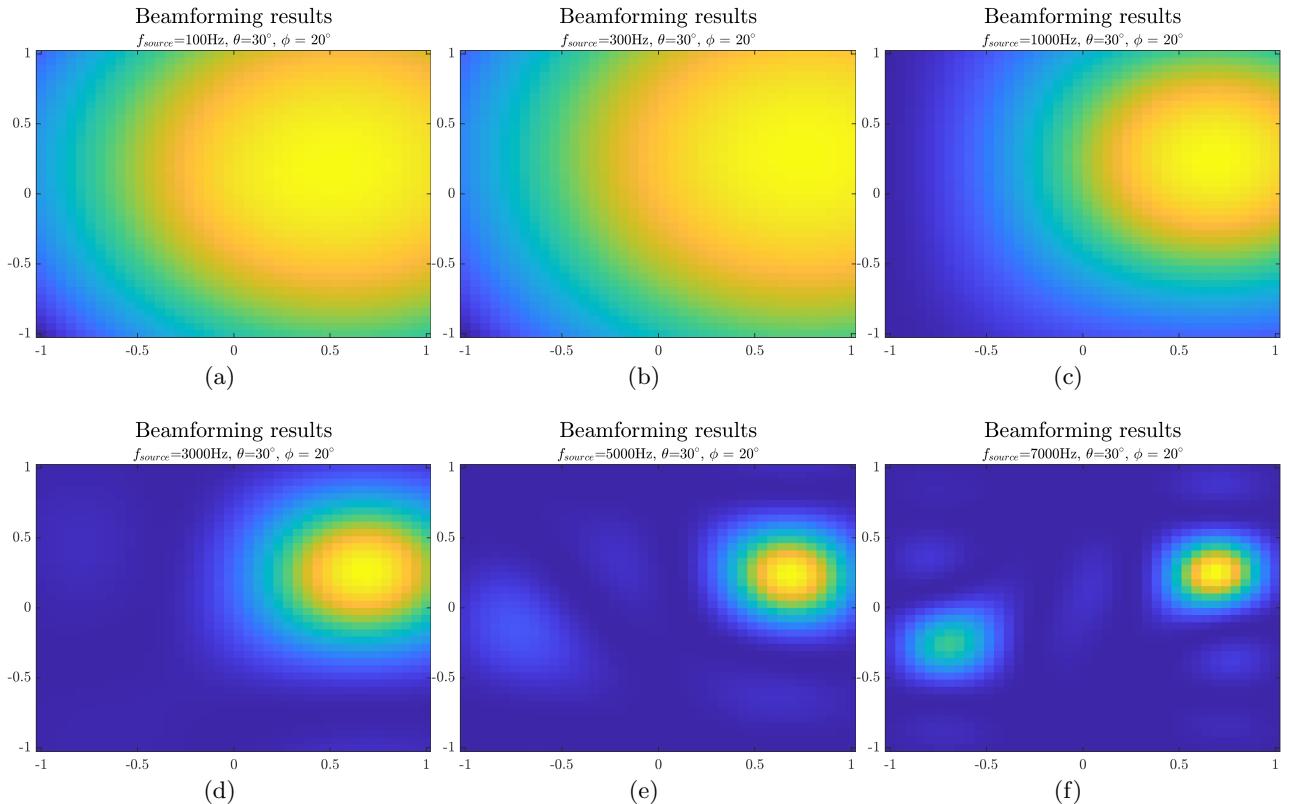


Figure 22: Beam forming results with one active array and one source.

It is clear from figure 22 that the beam forming algorithm is showing more define results at higher frequencies, as expected. However, mirroring effects can be seen, along with side lobes, at higher frequencies. The mirroring originates from the phase-shift method that becomes less accurate at higher frequencies.

4.3.2 One source, four arrays

Here four arrays are active and spaced with $d_{array} = 60$ mm, as illustrated in figure 8 and described in section 2.2.4. The signals are again sampled at a frequency of $f_{sample} = 15\,625$ Hz, and the source is placed 20 m away

from the array, at an angle of $\theta = 30^\circ$ and $\phi = 20^\circ$, as in the previous section. The results are presented in figure 23. The results are more precise, compared to when only one array is active.

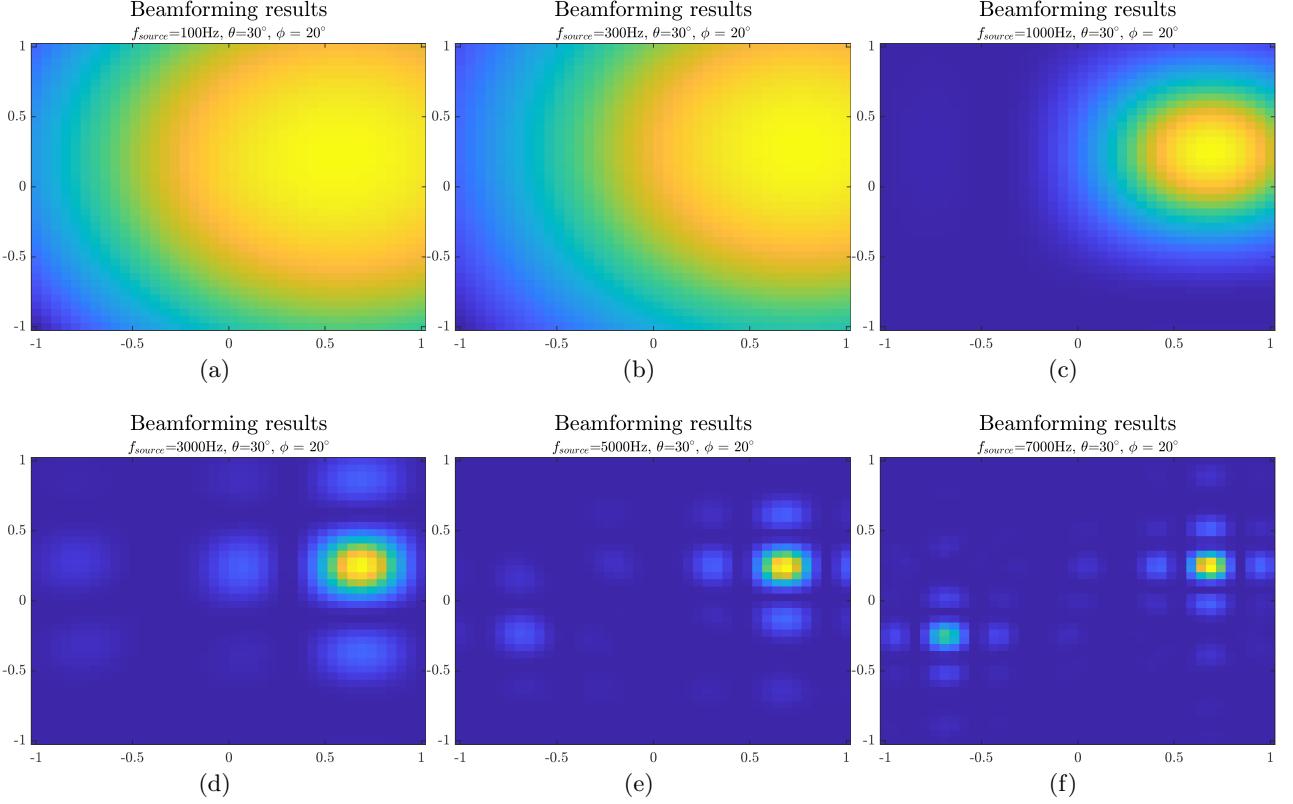


Figure 23: .

4.3.3 Two sources, one array

The performance of the beam forming with two sources is shown in this section. The first source is placed with an angle of $\theta_1 = 30^\circ$ and $\phi_1 = 20^\circ$, and the second source with an angle of $\theta_2 = -50^\circ$ and $\phi_2 = 20^\circ$. The location of the sources is illustrated in figure 24. Results are presented in figure 25. When the sources emit signals with lower, and different, frequencies the algorithm can not separate the two sources, see figure 25a. In figure 26 the same setup as in figure 25a is used, but evaluated at each separate frequency band. Here, it can be seen that the beam forming can find the two different sources before the signals from the different frequency bands have been added together.

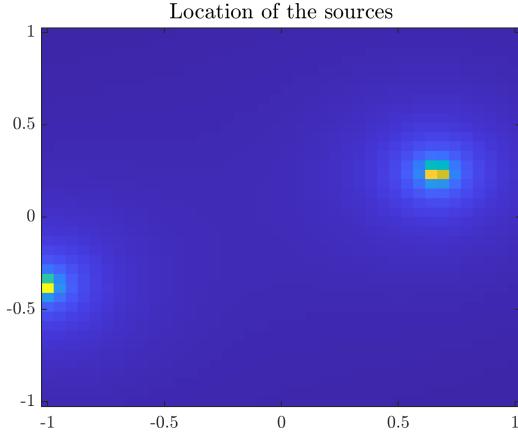


Figure 24: Actual location of two sources.

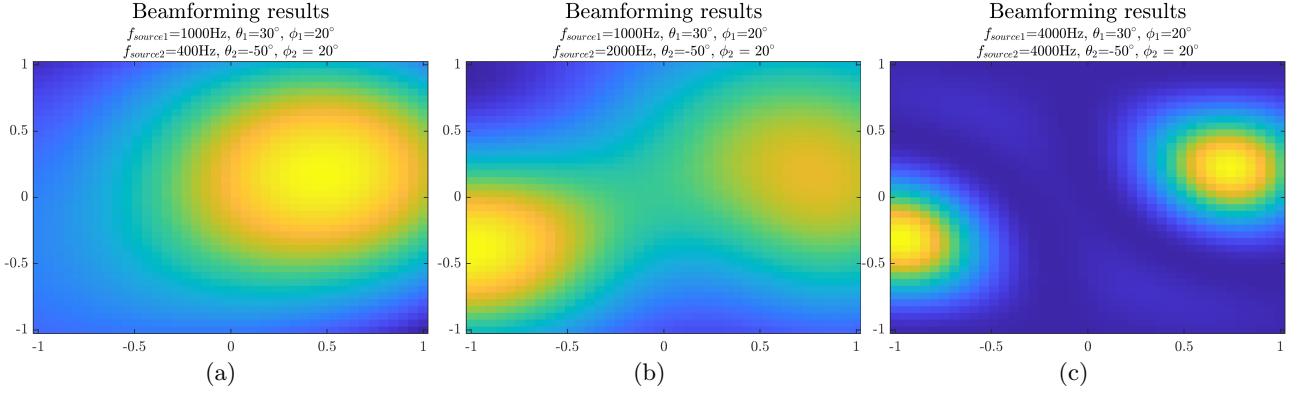


Figure 25: Beam forming results with two sources and one array. (a) at $f_{source1} = 1000$ Hz and $f_{source2} = 300$ Hz, (b) at $f_{source1} = 1000$ Hz and $f_{source2} = 2000$ Hz, (c) at $f_{source1} = 4000$ Hz and $f_{source2} = 4000$ Hz

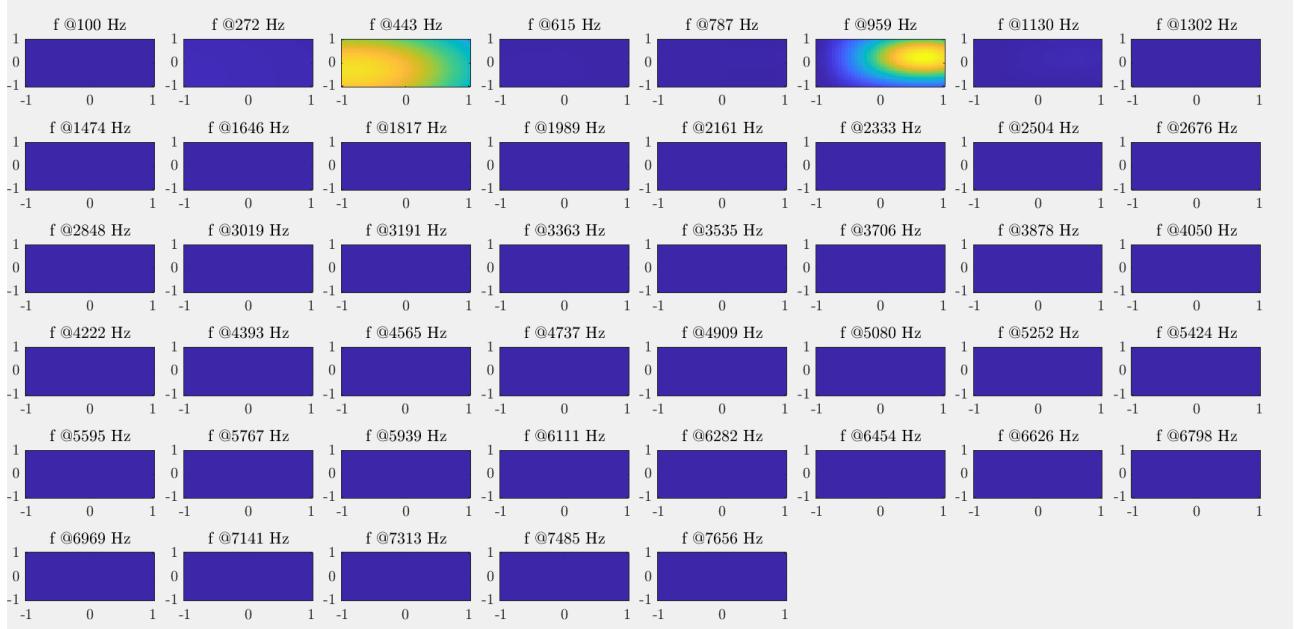


Figure 26: Beam forming results from each separate frequency, for the same case as in figure 25a two sources at $f_{source1} = 1000$ Hz and $f_{source2} = 400$ Hz.

4.3.4 Two sources, four arrays

Same simulations with two sources, with the same placement as in section 4.3.3 and figure 24, have been performed here with four arrays. The results can be found in figure 27. Here, it can be seen that signals from four arrays makes it possible to separate the two sources that emits lower frequencies (figure 27a) better than one array (figure 25a)

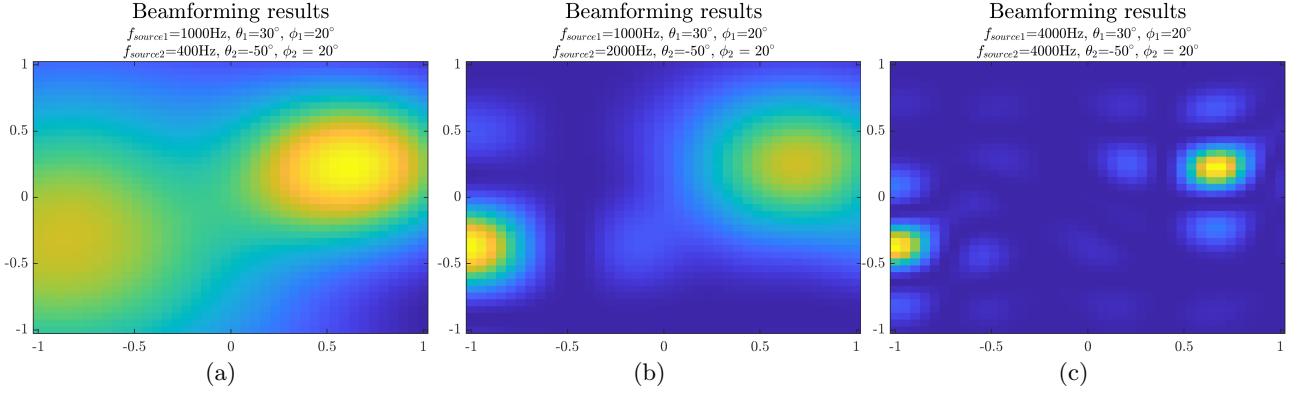


Figure 27: Beam forming results with two sources and four arrays. (a) at $f_{source1} = 1000\text{ Hz}$ and $f_{source2} = 300\text{ Hz}$, (b) at $f_{source1} = 1000\text{ Hz}$ and $f_{source2} = 2000\text{ Hz}$, (c) at $f_{source1} = 4000\text{ Hz}$ and $f_{source2} = 4000\text{ Hz}$.

4.3.5 Higher sampling frequency

The mirroring phenomena that occurs at higher frequencies, for example seen in figure 22f and 23f, is not present when sampling at a higher frequency. The results with $f_{sampling} = 48\,828\text{ Hz}$ is presented in figure 28. Here, no mirroring is seen and only the side lobes are present.

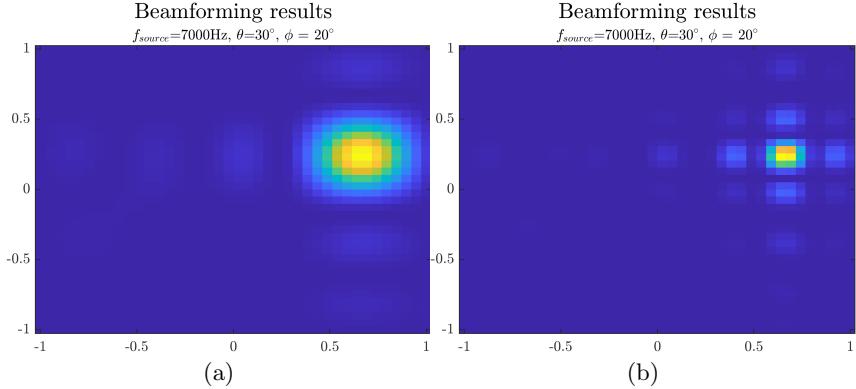


Figure 28: Beam forming when the signals have been sampled with $f_{sampling} = 48\,828\text{ Hz}$ from one source at $f = 7000\text{ Hz}$ and (a) one array, (b) four arrays.

5 Real data

When testing the beam forming algorithm on real data there are many aspects that affects the results, where the spacing between elements, and number of arrays have a large impact on the results, as seen when the algorithm was tested on emulated data. Apart from hardware, reflections and noise from the surrounding environment is predicted to highly influence the the performance. All results presented in this section have been recorded in Studion at Saab. The specifications of the array is presented in table 4. Unfortunately, the exact angle at which the source was placed is unknown, since there where no equipment available.

Table 4: Array specifications.

| Arrays | \mathbf{r}_a | columns | rows | uniform distance (m) | $f_{sampling}$ |
|--------|----------------|---------|------|----------------------|----------------|
| 1 | (0,0) | 8 | 8 | 0.02 | 48 828 Hz |

5.1 Sine waves

First, a sine wave was emitted with a frequency of $f = 3\text{ kHz}$ from one source placed at approximately $\theta = 30^\circ$ and $\phi = 0^\circ$ (50° from the normal of the array). The result is presented in figure 29, and seems to bee in agreement with where the source where placed.

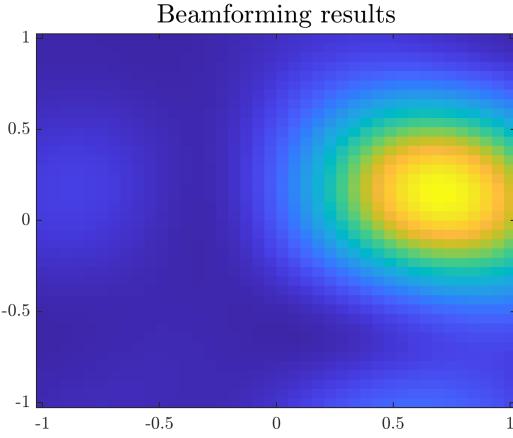


Figure 29: One source, emitting a sine wave of $f = 3\text{ kHz}$.

Then, two sources where placed at $\theta_1 = 30^\circ$, $\phi_1 = 0^\circ$ and $\theta_2 = 30^\circ$, $\phi_2 = 180^\circ$, both with a frequency of $f = 3\text{ kHz}$. The result is presented in figure 30. Only one source have been detected by the beam former.

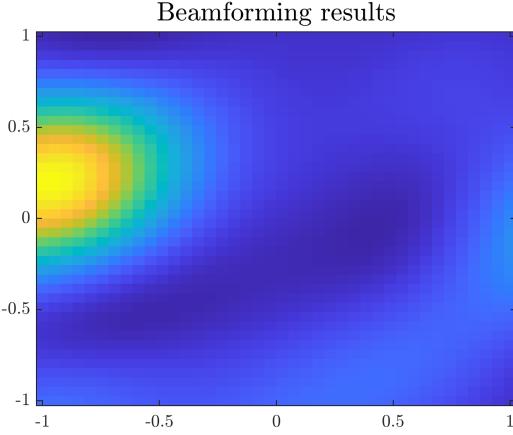


Figure 30: Two sources, both emitting a sine wave of $f = 3\text{ kHz}$.

5.2 Music

Three separate measurements of music were done. Here, two sources where placed at $\theta_1 = -30^\circ$, $\phi_1 = 0^\circ$ and $\theta_2 = 30^\circ$, $\phi_2 = 180^\circ$. In all measurements, different songs where played by the two sources. The results are found in figure 31.

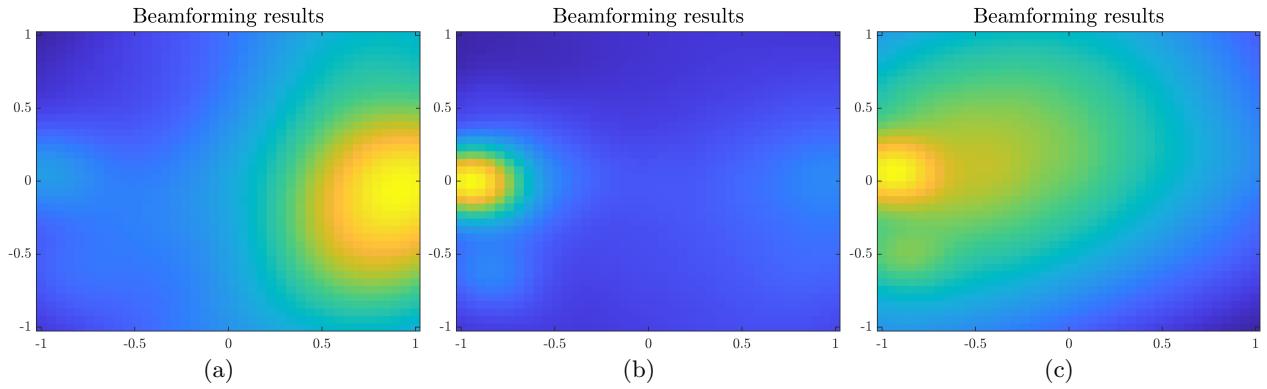


Figure 31: Three different recordings of music, emitted from two different sources.

The sources was found by the beam forming algorithms at $\theta = 39^\circ$ and $\phi = 0^\circ$ in figure 31a, at $\theta = 40^\circ$ and $\phi = 177^\circ$ in figure 31b and at $\theta = 38^\circ$ and $\phi = 177^\circ$ in figure 31c. Again, only one source was detected by the beam former (as for the case a sine wave from two sources), and was found further to the sides than where they where placed in reality.

It have not been established why the beam former is unable to find two sources. It have been noted, based on results from emulated data, that the algorithm is sensitive to difference in frequency from two sources. This could explain why the performance on on music is lower, but not why only one source is detected when the two signals are sine waves of equal frequency. Additionally, four microphones are out of phase, and lower in amplitude, than the rest of the microphones.

6 Future developments

6.1 Surrounding environment, reflections

Reflections from the surrounding environment can be an issue when analyzing the microphone array. Walls are interacting the same way with sound waves as a perfect electric conductor (PEC) are interacting with EM waves. Therefore, a smaller room can be interpreted as a waveguide and cause reflections, standing wave patterns etc. In the current algorithm, it is assumed that the sound reaches the microphones without any reflections. In order to obtain more accurate results in a room with reflecting walls, tables etc, the algorithms needs to be developed further.

6.2 Hardware

Ideally, the directionality of one microphone is isotropic. When mounted on a PCB, the sound will reach the microphones through holes in the PCB and the noise field might no longer be isotropic. This have not been taken into account in the current algorithm when processing the signal algorithms, but rather assumed that the microphones have isotropic directionality. Other factors, such as non-ideal anti-aliasing filters in the microphones, can have an effect on the performance of the beam forming algorithms. The microphones should be able to be sampled at different frequencies, and therefore the built-in anti-aliasing filter should adapt to the current sampling frequency. However, this does not seem to be the case when the microphones were tested with a sine wave of 8500 Hz at a sampling frequency of 15 625 Hz. Even though we can choose to not record any sound at frequencies above $f = 2f_{sampling}$, noise with frequencies above this will still be present in the signals.

6.3 Adaptive array configuration

Since the arrays are rectangular it will be possible to stack several arrays both in the horizontal and vertical direction. This will enable the possibility to develop an adaptive array configuration of elements that yield in optimal performance, in a similar way as discussed with one (or four) arrays in a previous section.

6.4 Calibration

Amplitude and phase calibrations of the microphones have not been performed, and is something that should be done in the future. Mismatch in amplitude of different microphones could be frequency dependent, so the frequency response of those can be used to make sure that the amplitude of all microphones are the same. Calibration signal measurements should be performed in an anechoic chamber in order to record clean signals, to avoid reflections.

6.5 Phase shift

The current phase-shift method is based on the derivative of the signal, and therefore dependent on both the previous and next sample, see equation (24). As the frequency of the signal is increasing the number of samples per period is decreasing, making the derivative of the signal, and therefore also phase-shifting, less accurate. When the frequency is at nyquist frequency, a mirroring effect is seen as previously shown. To get a better phase-shift at higher frequencies other methods have been tested, such as phase-shift with the use of Hilbert transformations. However, other methods may have longer computation times, and is therefore something that can be investigated further.

Another, very interesting aspect of phase-shift of signals is to perform phase-shift independent of frequency. The phase-shift implemented in this project is dependent on frequency, and is therefore performed at a chosen number of frequency bands for every microphone element.

6.6 Tapering

Hamming window osv

6.7 DDC and DUC

Digital down conversion (DDC) of a signal can enable faster computation time, since there is less data to process. A consequence of DDC of a real signal is that it becomes I and Q data. This type of data is relatively easy to phase-shift. An alternative would be to perform DDC and band pass filtering, for the frequency dependent phase-shift, at the same time. This was investigated, but not with enough success to develop the method further, and could therefore be something to look more into.

6.8 Linear phase filter

A FIR filter that introduces a phase shift proportional to the frequency at a proportionality rate K as well as little to no amplitude change of the filtered signal, would be highly desirable. The phase shift that is implemented to each signal in the array, is proportional to the frequency. With this filter, it is possible to only filter the signal coming from each element once. The only parameter that has to change is the proportionality rate of the phase, of the FIR filter.

6.9 Distance measurement

If the array antenna is sufficiently large compared to the distance to the source in terms of wavelengths, it is possible to determine the distance to the source. This is possible if the incoming waves are sufficiently spherical. In that case the plane wave approximation is not used. Instead, it is possible to test for a distance. Instead of having a 2D grid of θ and φ angles, we have a 3D grid with both angles and distances.