Speech Enhancement using Multiple Transducers

Craig Anderson

A Thesis submitted to the Victoria University of Wellington in fulfilment of the requirements for the degree of Master of Engineering

Victoria University of Wellington
2013

Abstract

In this thesis, three methods of speech enhancement techniques are investigated with applications in extreme noise environments.

Various beamforming techniques are evaluated for their performance characteristics in terms of signal to (distant) noise ratio and tolerance to design imperfections. Two suitable designs are identified with contrasting performance characteristics — the second order differential array, with excellent noise rejection but poor robustness; and a least squares design, with adequate noise rejection and good robustness.

Adaptive filters are introduced in the context of a simple noise canceller and later a post-processor for a dual beamformer system. Modifications to the least mean squares (LMS) filter are introduced to tolerate cross-talk between microphones or beamformer outputs.

An adaptive filter based post-processor beamforming system is designed and evaluated using a simulation involving speech in noisy environments. The beamforming methods developed are combined with the modified LMS adaptive filter to further reduce noise (if possible) based on correlations between noise signals in a beamformer directed to the talker and a complementary beamformer (nullformer) directed away from the talker. This system shows small, but not insignificant, improvements in noise reduction over purely beamforming based methods.

Blind source separation is introduced briefly as a potential future method for enhancing speech in noisy environments. The FastICA algorithm is evaluated on existing data sets and found to perform similarly to the post-processing system developed in this thesis. Future avenues of research in this field are highlighted.

This thesis would not have been possible without the support from my supervisors, Mark Poletti and Paul Teal, whose knowledge in their areas of research proved valuable throughout this thesis. Tait Communications, for providing insight into real world issues involved in capturing audio in noisy environments. Finally, my friends and family for supporting me throughout the duration of my thesis.

Contents

Chapter 1

Introduction

1.1 Motivation

1.1.1 Speech Enhancement in Noisy Environments

The issue of recording speech in noise is a well researched issue which has been in development over much of the last century. In the last two decades, much of this focus has been applied to cellphone and teleconferencing technology as advances in processing power, manufacturing techniques and communication bandwidth have improved to the point where sophisticated algorithms for noise reduction and/or beamforming are implementable in real-time.

This thesis focuses on the development of a computationally efficient system to pick up near-field speech in extreme noise environments. There are two main problems to overcome — the first of which is to find a method of isolating near sources from far sources, and the second of which is to find a method of noise reduction which is capable of dealing with high noise levels and is computationally efficient. The overall goal

is to develop a beamforming plus post-processor system to adaptively remove noise.

The focus of this thesis will be separated into three basic research groups: beamforming — to solve the speech isolation problem; adaptive filtering — to solve the noise reduction problem in a computationally efficient manner; and blind source separation — a technique which could ideally perform both tasks simultaneously.

1.2 Existing Techniques

1.2.1 Spectral Subtraction

One of the simpler techniques for noise reduction is spectral subtraction [1] [2] [3]. This technique involves computing an estimate of the power spectrum of the noise component in a signal and removing this from the spectrum of the original signal. An assumption made in this technique is that the speech and noise are uncorrelated (which aside from reverberant environments is a reasonable assumption). An estimate of noise can be obtained via a number of techniques. Popular methods include the use of voice activity detection and secondary sensors placed far from the talker. Voice activity detection, a method used extensively in speech compression, can be used to find segments of noise in the signal from which an estimate of the power spectrum can be obtained. A secondary sensor — such as other microphones located far from the talker, can be used to obtain clean estimates of the power spectrum. This technique requires an additional microphone in the system, which may be more effectively used in beamforming techniques for noise reduction; or be located far enough away from the talker to introduce delay problems in

power spectrum estimation or packaging problems. Placing a secondary microphone far from the talker may be impractical in all scenarios.

Suppose an estimate of the noise spectrum is obtained through some method, noise reduction can be achieved by simply subtracting the noise spectrum from the signal spectrum, leaving an estimate of the original speech signal. Considering a signal $x(t) = s(t) + n(t)$ comprising of $s(t)$, the signal from the talker and $n(t)$, the background noise. Taking the Fourier transforms, the signal can be represented by magnitude and phase components,

$$\begin{aligned} X(\omega) &= S(\omega) + N(\omega) \\ &= \|S(\omega)\|e^{i\phi_s(\omega)} + \|N(\omega)\|e^{i\phi_n(\omega)} \end{aligned}$$

Suppose there is an estimate of the noise spectrum available, $\hat{N}(\omega)$, its magnitude $\|\hat{N}(\omega)\|$ can be used to filter the noise from the input,

$$\begin{aligned} \|Y(\omega)\| &= \|X(\omega)\| - \|\hat{N}(\omega)\| \\ &= \|S(\omega)\| + \|N(\omega) - \hat{N}(\omega)\| \\ &\simeq \|S(\omega)\| \end{aligned}$$

The phase information of the signal is unknown, however typically the phase information from the original noisy signal is used to synthesise the original signal.

$$Y(\omega) = \hat{S}(\omega) = \|S(\omega)\|e^{i\phi_x(\omega)}$$

An inverse Fourier transform is taken, leaving the cleaned signal.

The resulting signal will have an improved signal to noise ratio, however artefacts of the processing technique will appear. Errors in the noise estimate will manifest as 'musical' noise, spurious peaks in the spectrum

of the output resulting from the subtraction process. This noise can be detrimental to speech intelligibility.

Voice activity detection (VAD) methods require detection of speech, which requires a reasonable signal to noise ratio to begin with. In addition, in some environments, the background noise may itself contain speech, rendering the technique useless. The use of a secondary noise–only microphone also has the same constraint, with the additional packaging requirements for this project. The spectral subtraction technique is unsuitable for noise reduction in this project primarily due to the levels of noise involved. Obtaining a clean estimate of the noise spectrum and using this in a subtraction technique without removing components of the speech becomes quite difficult. Suppose the background noise was Gaussian white noise and the speech level was low enough such that its spectral components were not easily identifiable in the noise, any estimate of the noise would most likely include the speech components. Filtering using this estimate would severely degrade (or eliminate) the speech component in the output.

1.2.2 Wiener Filtering

Wiener filtering is another popular technique for noise reduction [4]. The technique relies on either estimates of both the clean speech and noise spectra or the signal to noise ratio. As the estimate of the clean speech spectrum and the noise spectrum is generally unavailable for the purposes of this thesis (very low SNR), adaptive methods are used instead.

1.2.3 Differential Arrays

Differential microphone arrays (also known as gradient microphones) have been analysed for their potential in noise reduction since the 1940s [5], [6], [7]. Early analysis showed that this type of array design showed significant rejection of interferers far from the microphone array.

More recently, designs incorporating differential arrays have been developed with applications in telecommunications [8] [9]. These techniques primarily concern distant talkers and the analysis presented is limited to this scenario. In this thesis, the primary focus is to record near-field signals — signals originating at a distance comparable to the microphone array size. However, since the underlying concept is similar for near talkers, this earlier work could provide some insight into the potential for noise reduction using differential arrays in this project.

1.2.4 Adaptive Filtering

Adaptive filtering is a technique used for noise reduction, particularly in the area of echo cancellation in telecommunications [10]. Various forms exist with varying computational complexity and convergence properties. Least squares filters (LMS, NLMS and RLS) are popular methods of adaptive filtering and have been studied extensively [11] [12].

1.2.5 Blind Source Separation

Another technique that is briefly considered in this thesis is instantaneous blind source separation (BSS). Blind source separation is a recently developed (in the last two decades) technique which exploits

high order statistics of signals to find an unmixing matrix which is able to (with some limitations) separate complex mixtures of signals [13][14][15][16]. BSS requires no or little knowledge of the system to function, which is advantageous to beamforming/adaptive filtering techniques which typically require some amount of knowledge of a system (the talker location or assumptions about noise locations, for example).

1.3 Thesis Contents

This thesis will investigate the speech enhancing properties of three main techniques; microphone beamforming, adaptive filtering and blind source separation.

1.3.1 Microphone Arrays and Beamforming

The second chapter will introduce basic beamforming definitions starting from the wave equation. A number of beamforming techniques are investigated and their performance assessed for capturing near-field signals and suppressing far-field interference.

1.3.2 Adaptive Filtering

The third chapter will introduce adaptive filtering and its application to noise reduction. The least squares adaptive filter will be introduced as well as an original modification to NLMS to ensure robustness in low noise situations. A number of different adaptive filtering techniques will be evaluated.

1.3.3 System Design

The fourth chapter will present a speech enhancing system using a beamforming and adaptive filter based post-processor. Various combinations of beamforming techniques will be simulated and results compared.

1.3.4 Blind Source Separation

The final chapter will briefly introduce blind source separation, in particular the FastICA algorithm and its possible use for speech enhancement in noisy environments.

Chapter 2

Microphone Arrays and Beamforming

This chapter discusses various microphone array and beamforming techniques used in acoustics, in particular, near-field sound recording. First, the equations governing simple sound waves are derived from the wave equation. Next, some basic performance properties of microphone arrays are introduced. Finally, a series of different beamformer designs are introduced and analysed for suitability to speech enhancement.

2.1 Preliminaries

This section introduces the basic concepts applicable to microphone beamforming that are used throughout this thesis. A quick derivation of the point source and plane wave equations used to model simple sound sources is presented. Delay and sum beamforming is introduced to explain the concept of microphone beamforming. Near-field gain is introduced as one of the relevant performance measures of a beamforming

array. Finally, white noise gain is introduced as a performance measure of robustness of the array to imperfections in microphone responses.

2.1.1 Symbol Definitions

Table of symbols used in this chapter and their definitions

c	Wave propagation velocity
f	Wave Frequency
k	Wavenumber ($\frac{2\pi f}{c}$)
t	Time
$\boldsymbol{x}$	Position vector
w_m	Microphone weights
h_m	Source-microphone transfer function
H	Transfer function matrix
p	Single microphone output
s	Total array output
G	Array gain
NFG	Near-field gain
WNG	White noise gain

2.1.2 Point Sources and Plane Waves

Point Sources

The analysis of the microphone arrays to be presented later (in sections 2.2 to 2.5) will be considered using the spherical wave (point source) model for sound sources near microphones, and the plane wave model for sound sources far from the microphones. Although the point source model will form the focus of this thesis, plane waves are introduced

as they are prevalent in beamforming literature, and are applicable in cases where sound sources lie far, relative to the size of the array, from the sensors.

Starting with the wave equation and considering spherical symmetry (ignoring angular dependence),

$$\nabla^2\Psi = \frac{1}{c^2}\frac{\partial^2\Psi}{\partial t^2} \tag{2.1}$$

$$\nabla_r^2\Psi = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\frac{\partial\Psi}{\partial r}) \tag{2.2}$$

$$= \frac{\partial^2\Psi}{\partial r^2} + \frac{2}{r}\frac{\partial\Psi}{\partial r} \tag{2.3}$$

The wave equation becomes

$$\nabla_r^2\Psi - \frac{1}{c^2}\frac{\partial^2\Psi}{\partial t^2} = 0 \tag{2.4}$$

It can be shown that the wave represented by the equation,

$$\Psi(r,t) = \frac{e^{ik(ct-r)}}{r} \tag{2.5}$$

is a solution to the wave equation.

$$\nabla_r^2\Psi = \frac{1}{r}\frac{\partial^2 e^{ik(ct-r)}}{\partial r^2} - \frac{2}{r^2}\frac{\partial e^{ik(ct-r)}}{\partial r} + 2\frac{e^{ik(ct-r)}}{r^3} \tag{2.6}$$

$$+ \frac{2}{r}\left(\frac{1}{r}\frac{\partial e^{ik(ct-r)}}{\partial r} - \frac{e^{ik(ct-r)}}{r^2}\right) \tag{2.7}$$

$$= \frac{1}{r}\frac{\partial^2 e^{ik(ct-r)}}{\partial r^2} = \frac{-k^2}{r}e^{ik(ct-r)} \tag{2.8}$$

$$\frac{1}{c^2}\frac{\partial^2\Psi}{\partial t^2} = \frac{1}{c^2 r}\frac{\partial^2 e^{ik(ct-r)}}{\partial t^2} = \frac{-k^2c^2}{c^2 r}e^{ik(ct-r)} = \frac{-k^2}{r}e^{ik(ct-r)} = \nabla_r^2\Psi \tag{2.9}$$

Thus the wave equation is satisfied for the function

$$\Psi(r,t) = \frac{e^{ik(ct-r)}}{r} \tag{2.10}$$

A factor of $\frac{1}{4\pi}$ is typically included for energy conservation purposes, but is ignored in this thesis.

Plane Waves

Most of the literature on acoustic beamforming makes the assumption that the near-field condition can be relaxed. In this scheme, the sound pressure is modelled using plane waves propagating along an axis.

Starting with the one dimensional wave equation,

$$\frac{\partial^2 \Psi}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 \Psi}{\partial t^2} \tag{2.11}$$

Trying the plane wave solution,

$$\Psi = e^{ik(ct-x)} \tag{2.12}$$

$$\frac{\partial^2 \Psi}{\partial x^2} = -k^2\Psi \tag{2.13}$$

$$= \frac{1}{c^2}\frac{\partial^2 \Psi}{\partial t^2} \tag{2.14}$$

$$= \frac{-k^2c^2}{c^2}\Psi = -k^2\Psi \tag{2.15}$$

the wave equation is satisfied.

2.1.3 Preliminaries: Beam/Nullforming

Delay and Sum Beamforming

A common technique used in array design is delay and sum beamforming. In this method, directivity can be achieved by weighting each microphone in such a way that the incident wavefronts are aligned from the perspective of the array. The output of the array is denoted as,

$$s(k,r) = \sum_{m=1}^{M} w_m(k)h_m(k,r) \tag{2.16}$$

where w_m is the 'weight' — a complex (in general) number multiplied to the signal received by the m^{th} microphone in the array; and h_m denotes

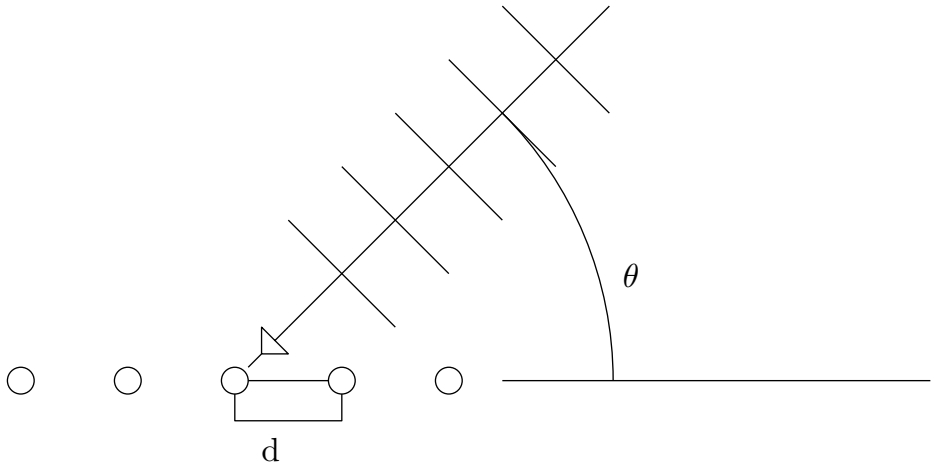


Figure 2.1: Plane wave incident on a microphone array.

the acoustic transfer function — the function describing the attenuation and phase shift of the audio source as the wave travels to the microphone, which can be represented by the Ψ functions earlier (equations 2.10, 2.12).

The array output can be expressed using vector notation as,

$$s(r) = \boldsymbol{w}^H \boldsymbol{h} \tag{2.17}$$

by ignoring the wavenumber for simplicity.

Consider a single pulse from a distant source incident on a line of microphones (Figure 2.1) and assuming the source is sufficiently far from the microphones such that its wavefronts are planar. The pulse will arrive at the microphones at different times depending on the angle between the wavevector and the vector formed by the microphone array. If the wavevector is perpendicular to the array vector, the time difference will be zero, if the wavevector is parallel, the time difference will be maximised, as the pulse now takes the largest possible time to prop-

agate along the array. To maximise the signal received from the microphone array due to the pulse, the microphones along the array must be weighted in such a way that they sample the same position of the pulse at the same time. This can be acheived by computing the additional distance the wavefront needs to travel to reach each microphone in the array and using this value to either time-delay the element or apply a phase shifted weight to the element.

2.1.4 Preliminaries: Near-field Gain

The near-field gain [17] is defined as the distance scaled ratio of the array output due to a source close to the array against a source in the far-field.

$$G(r_{\mathrm{n}}) = \frac{r_{\mathrm{n}}}{r_{\mathrm{f}}} \frac{s_{\mathrm{n}}}{s_{\mathrm{f}}} \tag{2.18}$$

For example, a single omnidirectional microphone has an output given by

$$s(k,r) = w(k)h(k,r) = w(k)\frac{e^{-ikr}}{r} \tag{2.19}$$

Its near-field gain is

$$\|G\| = \|\frac{r_{\mathrm{n}}}{r_{\mathrm{f}}} \frac{r_{\mathrm{f}} w(k) e^{-ikr_{\mathrm{n}}}}{r_{\mathrm{n}} w(k) e^{-ikr_{\mathrm{f}}}}\| = \|\frac{w(k)e^{-ikr_{\mathrm{n}}}}{w(k)e^{-ikr_{\mathrm{f}}}}\| = 1 \tag{2.20}$$

which provides a basis for one of the comparisons of arrays. A good microphone array will have greater than unity near-field gain across the frequency range of interest (for speech up to 3-4kHz, at least). This corresponds to greater rejection of far-field sources than a simple single microphone system. Since most noise sources considered in this project are far-field in nature, a high near-field gain array suggests good noise rejection.

2.1.5 Preliminaries: White Noise Gain

Tolerance to imperfections is an important property of a beamformer. Typically the microphones in the array will have slight differences in responses relative to the others as a result of intrinsic properties (frequency response not flat for example) or array placement errors. The objective of good array design is to ensure that these errors do not adversely affect the response of the array. A measure of this ability is known as white noise gain [18], a model of the response of the array due to these errors, modelled as Gaussian white noise. This white noise model operates under the assumption that the errors in array design or microphone mismatch are Gaussian in nature.

In general, the weights applied to each microphone (at a frequency $\omega = ck$) can be expressed as a complex number with an amplitude and phase.

$$w(k) = Ae^{i\phi} \tag{2.21}$$

If there are imperfections in the microphone then the actual weight required to produce the desired response is given by

$$w_a(k) = A_a e^{i\phi_a} \tag{2.22}$$

(and is unknown)
This can be re-written as the product of the designed perfect knowledge weight multiplied by an error term representing the amplitude

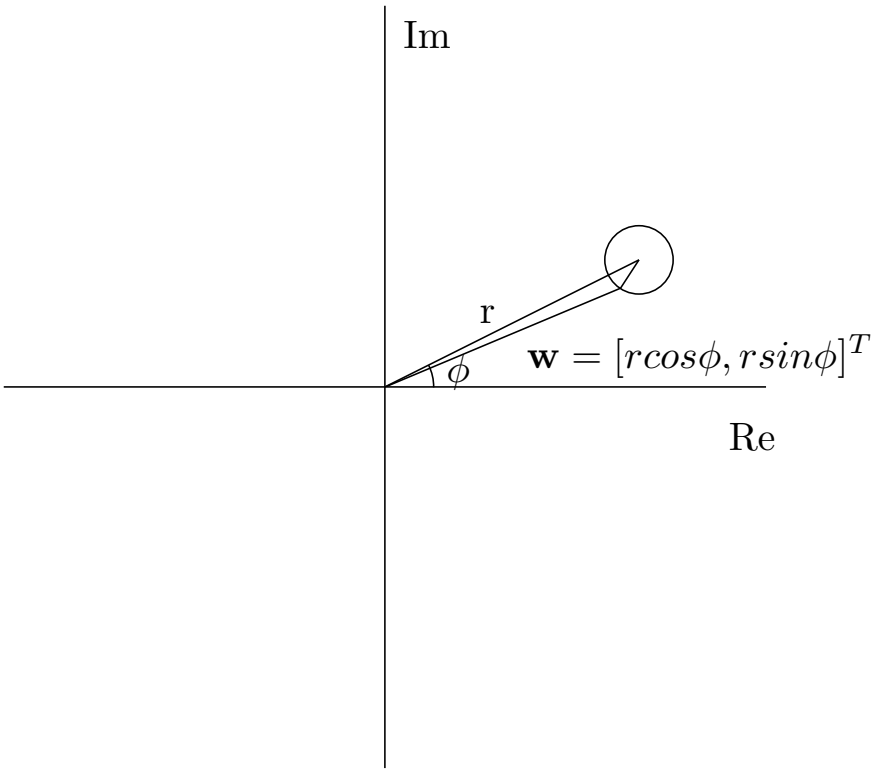


Figure 2.2: Graphical description of error in the weight vector.

and phase error of the microphone at a given frequency.

$$w_a(k) = Ae^{i\phi}[\frac{A_a}{A}e^{i(\phi_a-\phi)}] \tag{2.23}$$

$$= Ae^{i\phi}[\alpha e^{i\Delta\phi}] \tag{2.24}$$

$$= w(k)e(k) \tag{2.25}$$

$$e(k) = \frac{A_a}{A}e^{i\Delta\phi} \tag{2.26}$$

As seen in Figure 2.2, the error in the weight vector could be described

as lying with high probability inside a circle of some radius about the 'true' vector.

The white noise gain is then defined as the array gain for some near-field source over the array gain for isotropic white noise — the model for the error in the weight vectors. This can be quantified by finding the gain due to the source over the gain due to isotropic white noise. The white noise gain can be expressed as,

$$WNG = \frac{\boldsymbol{w}^H R_{\mathrm{n}} \boldsymbol{w}}{\boldsymbol{w}^H R_{\mathrm{f}} \boldsymbol{w}} \tag{2.27}$$

where R_{n} is the spatial autocorrelation matrix between the source and the microphones — the outer product of $\boldsymbol{h}$ and $\boldsymbol{h}^H$ from (2.17), $\boldsymbol{w}$ is the vector containing the microphone weights and R_{f} is the far-field correlation matrix — which for isotropic white noise is the identity matrix.

2.2 Differential Arrays

In this section, differential microphone arrays will be introduced and analysed. The equations governing theoretical performance of first and second order differential arrays will be derived in terms of point sources, along with their performance characteristics — frequency response, near-field gain and white noise gain.

2.2.1 Introduction

A technique for improving speech intelligibility is to create a microphone array with a highly directional response. A simple method of achieving this is to create a differential array. A differential array (also known as a gradient microphone) consists of at least two microphones closely spaced to exploit pressure gradients [7]. The technique exploits the spatial sampling nature of the array — each microphone samples the pressure field at the microphone position. For a high frequency wave the samples received will show large differences in amplitude relative to one another, producing a large response from the differential array. For sources close to the array, the sampled values would exhibit attenuation depending on the distance from the source to each microphone in the array. Most of the existing literature on differential arrays refers to far-field microphone arrays (where the rapid wave attenuation effects are not exploited), typically used for teleconferencing applications where the talker is far from the array [8].

2.2.2 First and Second Order Arrays

The near-field enhancement produced by first and second order differential arrays is derived as follows:

Starting with the equation for a point source (2.10),

$$p(k, \boldsymbol{x}, \boldsymbol{x}_m) = \frac{e^{-ik\|\boldsymbol{x}-\boldsymbol{x}_m\|}}{\|\boldsymbol{x}-\boldsymbol{x}_m\|} \tag{2.28}$$

Let $r = \|\boldsymbol{x} - \boldsymbol{x}_m\|$

$$p(k, r) = \frac{e^{-ikr}}{r} = f(r)g(r) \tag{2.29}$$

where $f(r) = e^{-ikr}$ and $g(r) = r^{-1}$.

Considering the differential behaviour along the microphone array axis x and applying the chain and product rules,

$$\begin{aligned}\frac{dp}{dx} &= \frac{dp(k,r)}{dr}\frac{dr}{dx} \\ &= \frac{df(r)}{dr}g(r)\frac{dr}{dx} + f(r)\frac{dg(r)}{dr}\frac{dr}{dx} \\ &= -ikf(r)g(r)\frac{x}{r} - f(r)g(r)\frac{x}{r^2}\end{aligned}$$

and noting that $\frac{x}{r} = \cos\theta$, the first order differential array gives an output,

$$p_d(k, r, \theta) = -p(k, r)(\frac{1}{r} + ik)\cos\theta \tag{2.30}$$

Compared with a single omnidirectional microphone, the first order differential array exhibits increased gain in the near-field. The relative on-axis gain for this array is $-(\frac{1}{r} + ik)$. The frequency dependence suggests first order high pass behaviour which can be corrected using a simple first order low-pass filter. The beam-pattern is now a dipole, showing complete rejection of noise arriving laterally.

Greater near-field gain and directionality can be attained using higher

order differential arrays. The second order array will be derived here (higher orders become mathematically tedious).

$$\begin{aligned} p_q(k,r,\theta) &= \frac{dp_d(k,r)}{dx} \\ &= -ik\left(\frac{df(r)}{dx}g(r)\frac{x^2}{r^2} + f(r)\frac{dg(r)}{dx}\frac{x^2}{r^2} + f(r)g(r)\frac{1}{r}\right) \\ &- \left(\frac{df(r)}{dx}g(r)\frac{x^2}{r^3} + f(r)\frac{dg(r)}{dx}\frac{x^2}{r^3} + f(r)g(r)\frac{1}{r^2}\right) \\ &= p(k,r)\left(-\left(\frac{ik}{r}+\frac{1}{r^2}\right) + \cos^2\theta\left(-k^2+\frac{3ik}{r}+\frac{3}{r^2}\right)\right) \end{aligned}$$

The second order array shows greater sensitivity to pressure variations along the axis than the first and zeroth order arrays. The directionality has improved along axis due to the $\cos^2\theta$ dependence. Like the first order design, high-pass behaviour is present as a combination of first and second order components. A second order low-pass filter can be used to correct this behaviour.

By considering the distance between the source and array, it is possible in some cases to ignore low-pass correction filters. At close distances the r^{-1} (first) and r^{-2} (second order differential array) terms dominate, neither of which contain frequency dependence.

$$p_d(k,r,\theta) \simeq \frac{-p(r)}{r}\cos\theta \qquad r \ll 1 \tag{2.31}$$

$$p_q(k,r,\theta) \simeq p(r)\left(\frac{3\cos^2\theta - 1}{r^2}\right) \qquad r \ll 1 \tag{2.32}$$

At high frequencies, the r^{-1} and r^{-2} terms lose significance and the frequency dependent terms dominate

$$p_d(k,r,\theta) \simeq -ikp(r)\cos\theta \qquad k \gg 1 \tag{2.33}$$

$$p_q(k,r,\theta) \simeq -k^2 p(r)\cos^2\theta \qquad k \gg 1 \tag{2.34}$$

Using these two sets of equations, it is possible to derive the transition frequency between flat frequency response and n^{th} order high-pass response. To simplify, assume that the talker is on-axis with the arrays ($\cos\theta, \cos^2\theta = 1$), the transition frequency/wavenumber occurs when the two equations are equal (taking absolute values):

$$\text{(first order)} \qquad \frac{1}{r} = k$$
$$\text{(second order)} \qquad \frac{2}{r^2} = k^2$$

$$k_d = \frac{1}{r} \tag{2.35}$$
$$k_q = \frac{\sqrt{2}}{r} \tag{2.36}$$

Existing speech transmission methods typically operate in a limited frequency range of between 300 and 3.5kHz which captures most of the energy in human speech. The talker would need to be located no more than 16.1mm (first order) and 22.7mm (second order) away from the microphone array in order to maintain flat response across the speech band. It is apparent that even a small perturbation ($\pm$10mm) in talker-array distance could have a significant effect on the frequency response of the array — the transition frequency between flat and high-pass response is a function of talker-array distance. For this reason, a low-pass correction filter to flatten the frequency response would be difficult to implement.

The use of this type of array for speech enhancement would require careful talker-array arrangement. A minimum distance could be implemented by mounting the microphones inside a box with some specified distance between the ends of the the array and the walls of the box. A

low-pass corrector could be designed for this minimum distance provided it is acceptable for some high-pass behaviour if the user moves away from the box.

The near-field gain of differential arrays can be calculated using the equations derived for the low and high frequency gain (the frequency transition equations), which additionally correspond to near and far response, and taking the absolute values. The low-frequency (or close distance) equations for the response of the differential arrays are as follows,

$$\begin{aligned} p_d &\simeq \frac{-p(k,r)}{r} && \text{if } r \ll 1 \\ \|p_d\| &\simeq \frac{1}{r_{\mathrm{n}}^2} \\ p_q &\simeq \left(\frac{2p(k,r)}{r^2}\right) && \text{if } r \ll 1 \\ \|p_q\| &\simeq \frac{2}{r_{\mathrm{n}}^3} \end{aligned}$$

The far distance (or high frequency) equations are as follows,

$$\begin{aligned} p_d &\simeq -ikp(r) && \text{if } r \gg 1 \\ \|p_d\| &\simeq \frac{k}{r_{\mathrm{f}}} \\ p_q &\simeq -k^2 p(r) && \text{if } r \gg 1 \\ \|p_q\| &\simeq \frac{k^2}{r_{\mathrm{f}}} \end{aligned}$$

Inserting these equations into the near-field gain equations shown in Section 2.1 (2.18), the near-field gain of the differential arrays can be

derived on-axis as

$$\begin{aligned} NFG_d &= \frac{r_\mathrm{n}}{r_\mathrm{f}} \frac{1}{r_\mathrm{n}^2} \frac{r_\mathrm{f}}{k} \\ &= \frac{1}{kr_\mathrm{n}} \\ NFG_q &= \frac{r_\mathrm{n}}{r_\mathrm{f}} \frac{2}{r_\mathrm{n}^3} \frac{r_\mathrm{f}}{k^2} \\ &= \frac{2}{k^2 r_\mathrm{n}^2} \end{aligned}$$

The differential arrays show $(kr_\mathrm{n})^{-n}$ near-field gain behaviour for a fixed talker distance r_n located on the microphone array axis. This corresponds to near-field gain for the frequency range in the flat frequency response region derived earlier.

2.2.3 Simulations

So far the theoretical properties of differential arrays have been derived. One of the assumptions made initially was that the spacing between the microphones was infinitesimally small, an unrealistic condition to allow the easy derivation of these properties. To investigate the effects of finite spacing, a simulation was developed with MATLAB in which microphones were spaced with small (c/f_s m) distances between them. A sample rate of 44.1kHz was selected for testing, corresponding to a maximum microphone separation distance of 7.8mm to ensure that the differential array could produce close to theoretical performance — in terms of beampattern shape and near-field gain, up to 22kHz. First and second order differential arrays were simulated.

The effect of placing finite spacing between two microphones is demonstrated in Figure 2.5. The theoretical case allows ever increasing re-

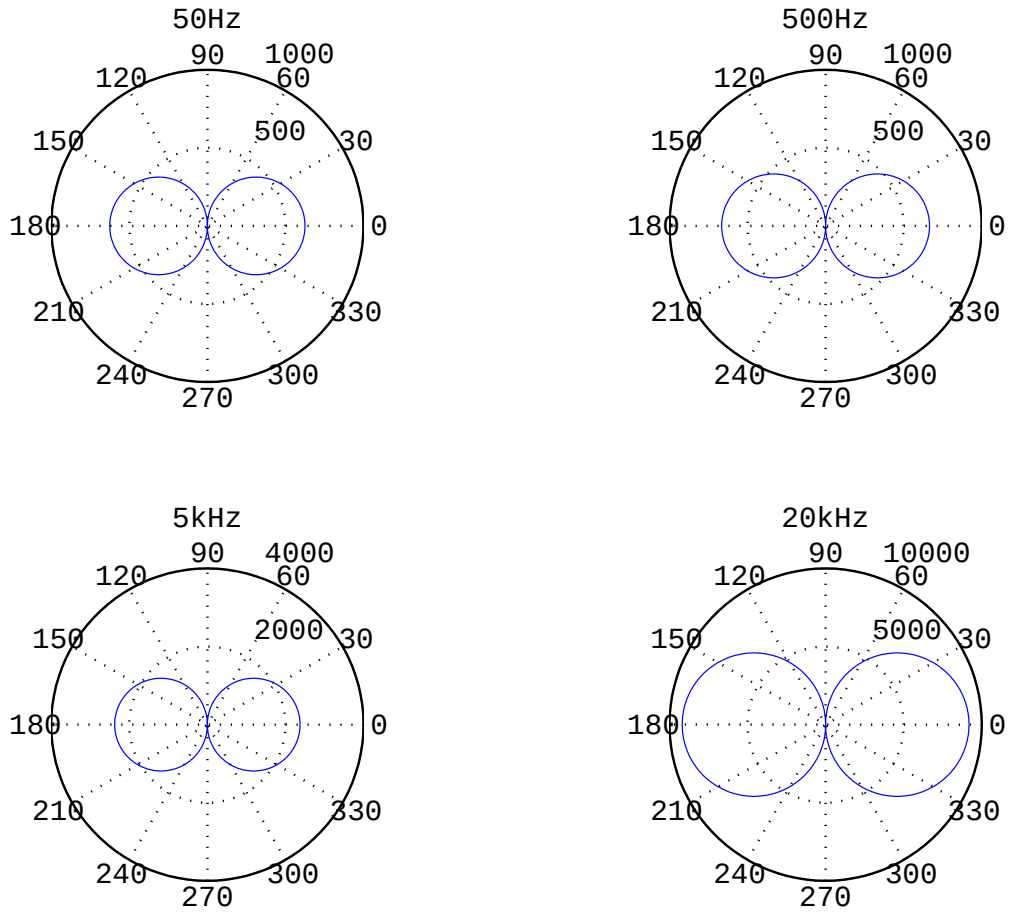


Figure 2.3: First order differential theoretical beampatterns at a distance of 4cm. The theoretical first order array produces a perfect dipole at all frequencies.

sponse with frequency as the gradient of an incident wave increases with frequency. In the finite spacing simulation, the frequency response still exhibits (near) flat response in the low frequency region and first order high pass behaviour up to the spatial Nyquist rate of the array. The spatial Nyquist rate arises from the sampling nature of the array. Each microphone samples the sound pressure at a regularly spaced interval, analogous to time domain sampling where samples are taken at regular time intervals — allowing perfect reconstruction of the signal up to

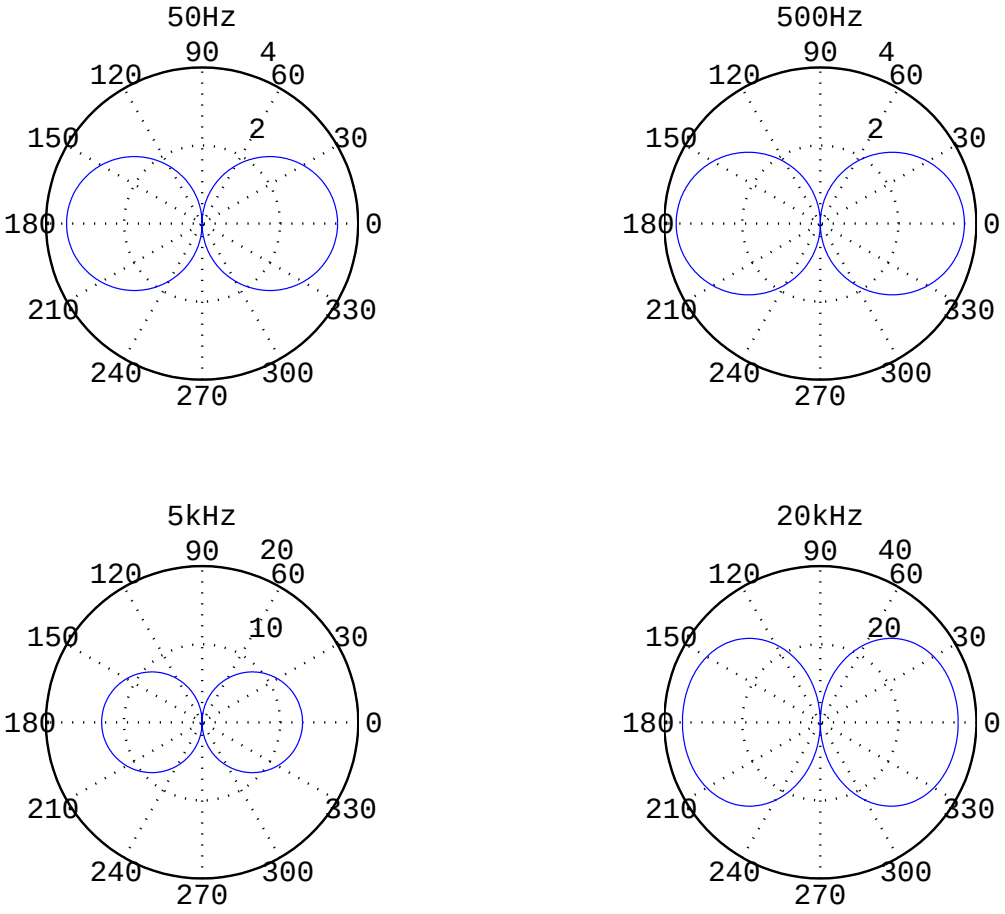


Figure 2.4: First order differential simulated array beampattern at a distance of 4cm. Discrete spacing produces slight differences from the theoretical case. The beampatterns are slightly distorted at higher frequencies.

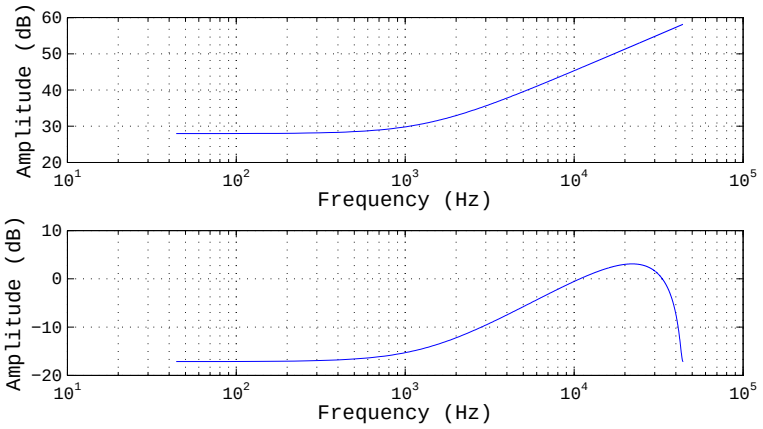


Figure 2.5: Infinitesimal spacing theoretical limit (top) and finite spacing simulation of the frequency response of a first order differential array. Note the high-pass behaviour at high frequencies.

the Nyquist rate (half the sampling rate). Spatial sampling dictates the ability of the array to resolve signal direction, the one of the key components of beamforming. The difference between infinitesimal sampling and discrete spacing is visible in the beampatterns (Figures 2.3 and 2.4). The discrete spacing first order array shows a slight difference in the shape of the beampattern at higher frequencies.

The second order response is illustrated in Figure 2.8. As in the first order case, the response is similar to the infinitesimal spacing response up to the Nyquist rate. Similarly to the first order array, the beampatterns show slight differences in shape at higher frequencies from the theoretical results (Figures 2.6 and 2.7).

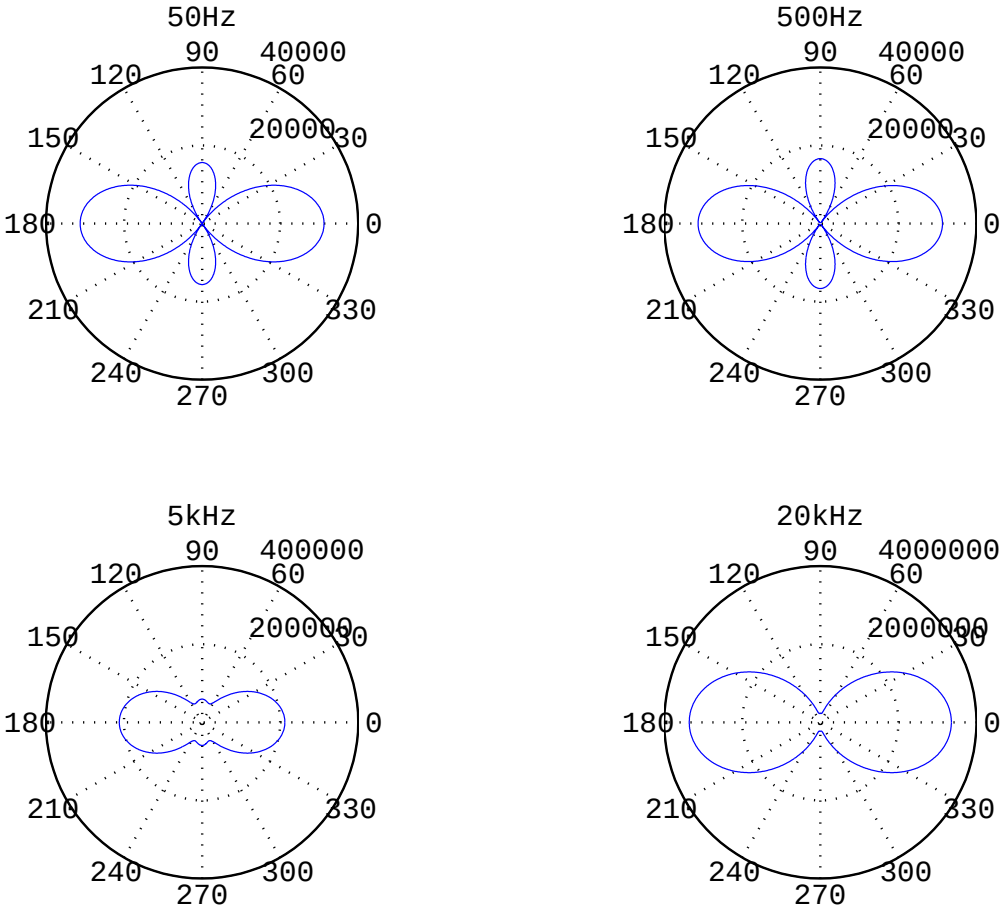


Figure 2.6: Theoretical response patterns for a source located 4cm away from a second order differential line array. At very high frequencies, the response to sources located laterally (90 or 180 degrees) is attenuated significantly.

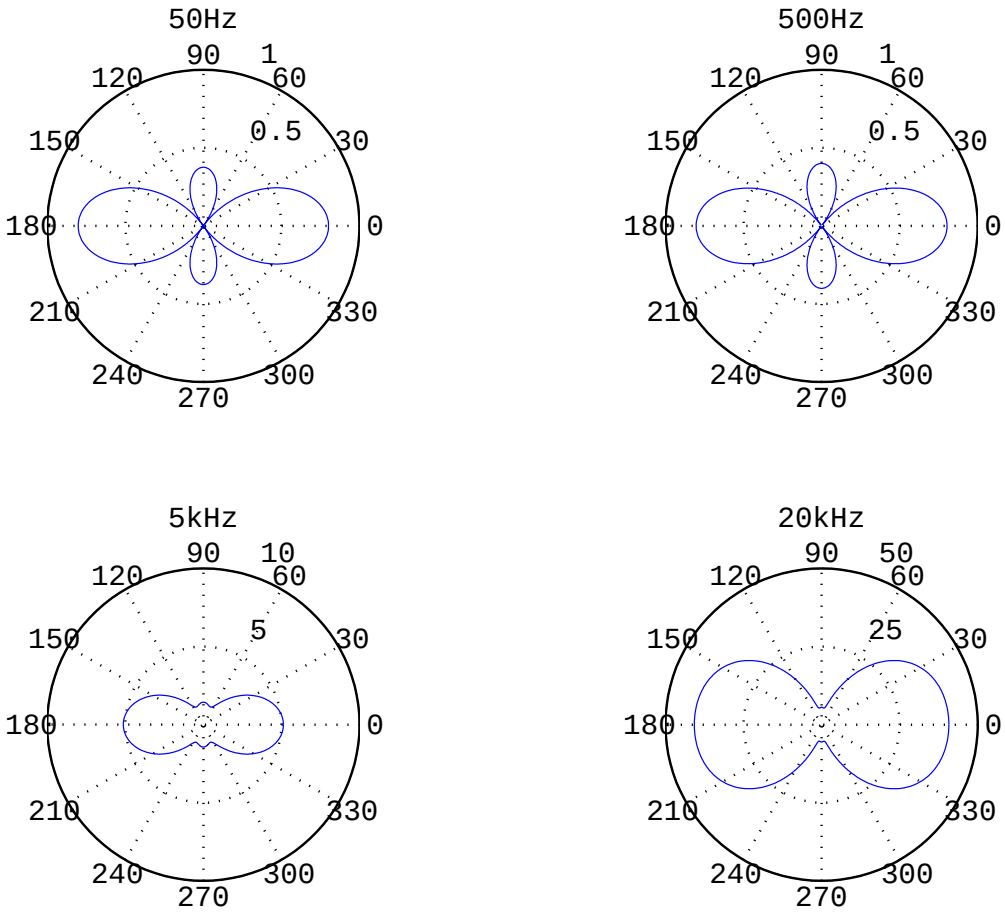


Figure 2.7: Simulated response for a source 4cm from a second order differential line array. Finite spacing introduces slight distortions to the theoretical response patterns presented in Figure 2.6.

The near-field gain for the simulated first and second order arrays is shown in Figure 2.9. Differential arrays show significant near-field gain at low frequencies (less than 1kHz), corresponding to good rejection of far-field interferers.

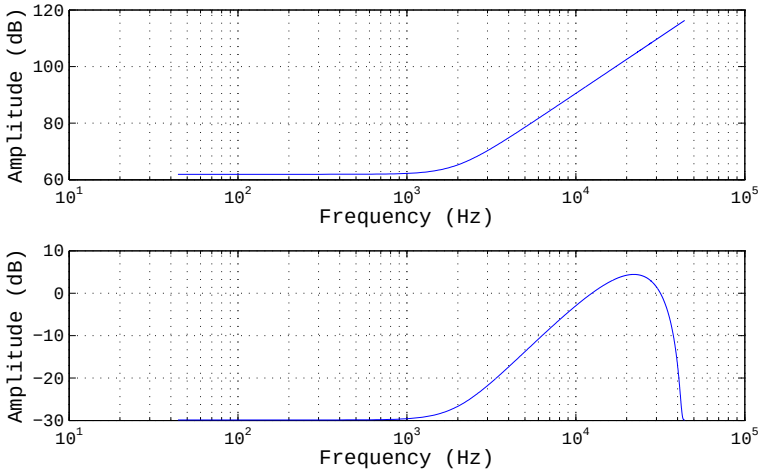


Figure 2.8: Frequency for the second order differential array, theory (top) vs. simulated finite spacing. Like the first order array, high-pass behaviour is present at high frequencies.

2.2.4 Summary

Differential arrays show promise for near-field enhancement of speech. However tolerance to microphone mismatch and high frequency boosting are significant issues which would need to be handled in a practical implementation. As seen in Figure 2.9, first and second order arrays show excellent rejection of far-field sources compared to single microphone solutions at low frequencies. The response patterns show almost complete rejection (first order — Figure 2.4) and attenuation (second order — Figure 2.7) of lateral noise sources, which when combined with its near-field gain behaviour (rejection of far sources), suggests that they could be ideal for speech enhancement for talkers close to the array. The

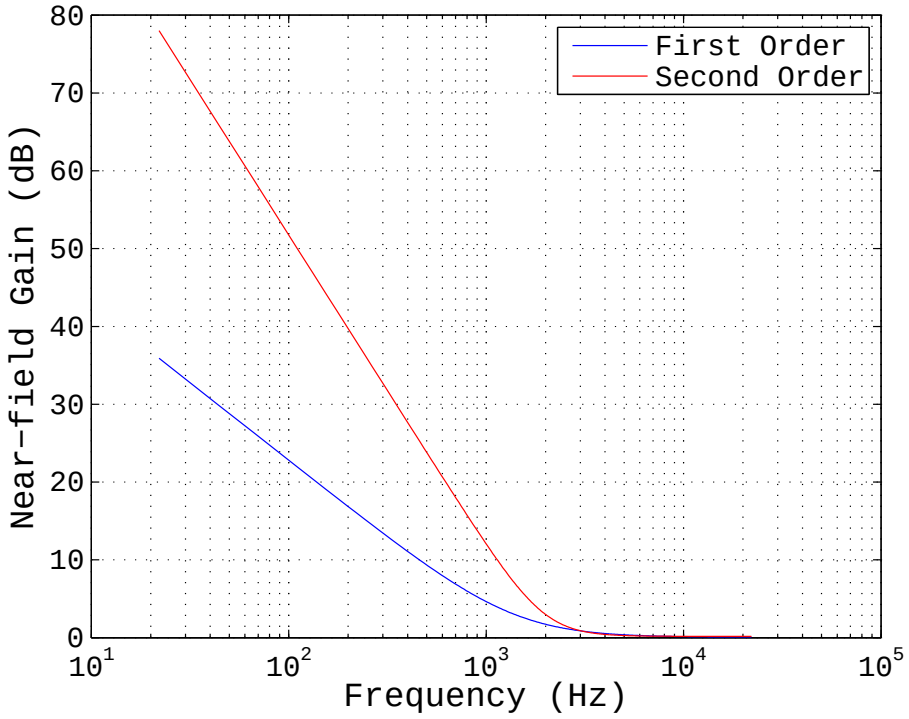


Figure 2.9: Near-field gain for the simulated first and second order differential arrays. The second order array shows significantly improved performance over the first order array.

main problem with differential arrays is the poor white noise gain in the speech band of the audio spectrum (Figure 2.10). The arrays show extreme sensitivity to microphone mismatch (or placement errors) which can severely degrade performance. The simplicity of the weight design does not allow robustness to be built in, as will be seen in the remainder of the chapter. A minor consideration is the high frequency boosting which occurs with differential arrays, however this can be easily corrected using either analog or digital low pass filtering.

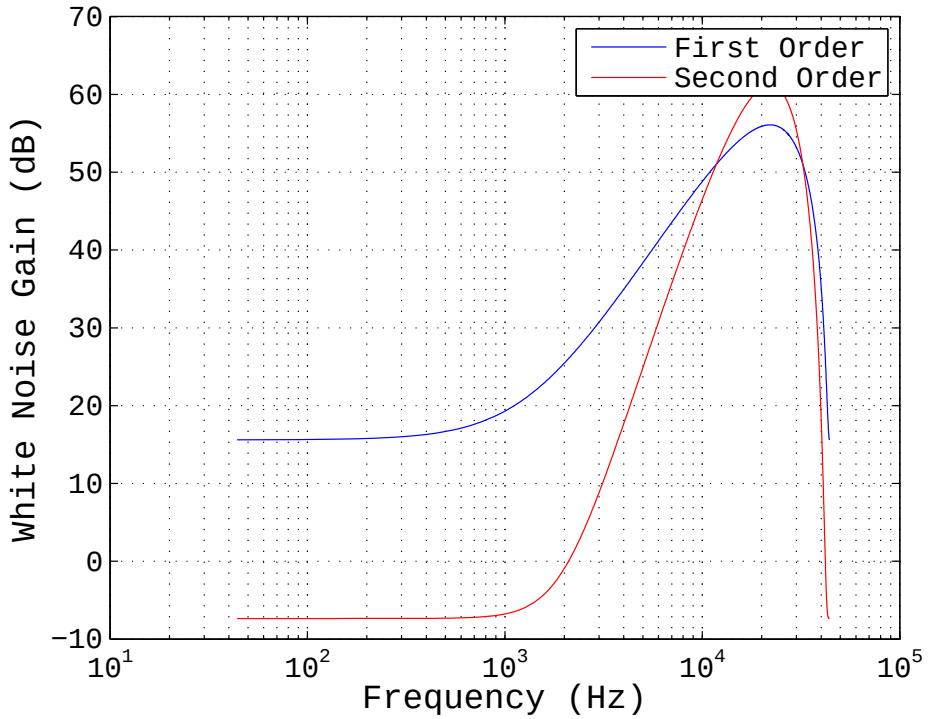


Figure 2.10: White noise gain for the simulated first and second order differential arrays. Second order arrays show very poor white noise gain at low frequencies, corresponding to the poor ability in handling microphone errors.

2.3 Maximising Near-field Gain

Another method of beamforming is to maximise the near-field gain at the talker location relative to the interferers, which can be assumed to be located in the far-field. This technique has previously been described in the context of reverberant environments where the microphone array receives a near-field signal plus reverberation assumed to be far-field in nature [17].

Consider a general M-element microphone array. The output of the array is given by

$$s_n(k, \boldsymbol{x}) = \sum_{m=0}^{M-1} w_m h_m(k, \boldsymbol{x}, \boldsymbol{x_m}) \tag{2.37}$$

$$h_m(k, \boldsymbol{x}, \boldsymbol{x_m}) = \frac{e^{-ik\|x - x_m\|}}{\|x - x_m\|} \tag{2.38}$$

where w_m is the microphone weighting, and h_m, the transfer function for microphone m, is the pressure due to a point source located at $\boldsymbol{x}$ incident on microphone m at location $\boldsymbol{x_m}$.

The objective is to find weights $\boldsymbol{w} = [w_0 w_1 \cdots w_{M-1}]^T$ which maximise the ratio between the signal received from the talker and the background noise,

$$\mu = \frac{E\{s_\mathrm{n}^H s_\mathrm{n}\}}{E\{s_\mathrm{f}^H s_\mathrm{f}\}} = \frac{\boldsymbol{w}^H R_\mathrm{n} \boldsymbol{w}}{\boldsymbol{w}^H R_\mathrm{f} \boldsymbol{w}} \tag{2.39}$$

Now, defining the near-field (talker) correlation matrix R_n as

$$R_\mathrm{n} = \boldsymbol{h}_\mathrm{n} \boldsymbol{h}_\mathrm{n}^H \tag{2.40}$$

where $\boldsymbol{h} = [h_0 h_1 \cdots h_{M-1}]^T$) — derived from (2.38).

The far-field (interferer(s)) correlation matrix R_f is defined as

$$R_\mathrm{f} = \boldsymbol{h}_\mathrm{f} \boldsymbol{h}_\mathrm{f}^H \tag{2.41}$$

where $\boldsymbol{h}_\mathrm{f}\boldsymbol{h}_\mathrm{f}^H$ is a matrix describing the average far-field interferer correlation.

If the constraint $\boldsymbol{w}^H R_\mathrm{f}\boldsymbol{w} = 1$ is introduced, then the method of Lagrange multipliers can be utilised to solve the problem.

$$L = \boldsymbol{w}^H R_\mathrm{n}\boldsymbol{w} + \mu(1 - \boldsymbol{w}^H R_\mathrm{f}\boldsymbol{w}) \tag{2.42}$$

$$\nabla_{\boldsymbol{w}} L = R_\mathrm{n}\boldsymbol{w} - \mu R_\mathrm{f}\boldsymbol{w} \tag{2.43}$$

$$0 = R_\mathrm{n}\boldsymbol{w} - \mu R_\mathrm{f}\boldsymbol{w} \tag{2.44}$$

$$R_\mathrm{n}\boldsymbol{w} = \mu R_\mathrm{f}\boldsymbol{w} \tag{2.45}$$

This is an example of a generalised eigenvalue problem $A\boldsymbol{w} = \mu B\boldsymbol{w}$, which can be solved using a number of methods. Provided B is invertible, the eigenvalues and corresponding eigenvectors can be solved through the eigenvalue decomposition of $Z = B^{-1}A$.

In order to obtain the optimal near-field gain, the average far-field spatial correlation matrix must be known. Using a Bessel function expansion of the point source model [19] and considering an infinite set of points located around the centre of the array at a distance of r_f, the average correlation between the i^{th} and j^{th} microphones is given by

$$\begin{aligned} R_{ij} &= \int\int p_i p_j^* \sin\theta d\theta d\phi \\ &= \int\int \sum_{n=0}^{\infty} \|h_n(kr_f)\|^2 j_n(kr_i) j_n^*(kr_j) \\ &\sum_{m=-n}^{n} Y_m^n(\theta_i,\phi_i) Y_m^{n*}(\theta_s,\phi_s) Y_m^{n*}(\theta_j,\phi_j) Y_m^n(\theta_s,\phi_s) \sin\theta d\theta d\phi \end{aligned}$$

where j_n is an n^{th} order spherical Bessel function of the first kind, y_n, is an n^{th} order spherical Bessel function of the second kind and Y_m^n, the spherical harmonics of order m, n.

The spherical harmonic terms can be simplified through the use of the spherical harmonic addition theorem [20] to obtain,

$$R_{ij} = \sum_{n=0}^{\infty} \|h_n(kr_f)\|^2 j_n(kr_i) j_n(kr_j)(2n+1) P_n(\cos\gamma_{ij}) \sin\theta d\theta d\phi \quad (2.46)$$

Now, using the identity [21],

$$\sum_{n=0}^{\infty} (2n+1) P_n(\cos\gamma_{ij}) j_n(kr_i) j_n(kr_j) = j_0(k\|r_i - r_j\|) \quad (2.47)$$

and noting that if r_f is very large $h_n(kr_f)$ reduces to

$$h_n(kr_f) \simeq (-i)^{n+1} \frac{e^{ikr_f}}{kr_f} \quad (2.48)$$

then the ij^{th} component of R is

$$R_{ij} = \frac{j_0(k\|r_i - r_j\|)}{r_f^2} \quad (2.49)$$

$$= \frac{\mathrm{sinc}(k\|r_i - r_j\|)}{r_f^2} \quad (2.50)$$

Since the near-field gain definition pre-multiplies the near and far-field components by the distance factors, R_{ij} can be simplified at this point to

$$R_{ij} = \mathrm{sinc}(k\|r_i - r_j\|) \quad (2.51)$$

The solution for maximising the near-field gain is

$$R_{\mathrm{f}}^{-1} R_{\mathrm{n}} \boldsymbol{w}_{\max} = \mu_{\max} \boldsymbol{w}_{\max} \quad (2.52)$$

Given R_{f} is a sinc matrix, the solution will be poorly conditioned at low frequencies ($\mathrm{sinc}(kd) \simeq 1$). Regularisation is therefore required to produce a robust solution (i.e., one with good white noise gain).

The regularised solution is found by inserting regularisation matrices into each of the near and far-field correlation matrices in (2.52).

$$(R_{\mathrm{f}} + r_f^2 \lambda I_M)^{-1} (R_{\mathrm{n}} + r_n^2 \lambda I_M) \boldsymbol{w}_{\max} = \mu_{\max} \boldsymbol{w}_{\max} \quad (2.53)$$

2.3.1 Simulations

The near-field gain array was tested using four microphones in a circular geometry. The diameter of the array was set to 7.8mm.

At low frequencies the unregularised maximal near-field gain method shows very poor white noise gain (solid line in Figure 2.11b) — indicating that the array would have difficulty handling far-field noise or distance/frequency mismatches between the microphones. Some near-field gain performance can be sacrificed in order to improve the white noise gain by introducing a regularisation matrix term into the spatial correlation matrices (2.53). As the regularisation parameter λ increases, the far-field correlation matrix tends to the identity matrix, converting the maximum near-field gain problem into a maximum white noise gain problem. Careful selection of the regularisation parameter is required to ensure that the array maintains both good near-field gain and white noise gain performance. In practise, this is difficult to do.

A regularisation parameter of $\lambda = 10^{-3}$ was inserted into (2.53) to design a more robust array. The result of introducing this regularisation parameter was an improvement in white noise gain throughout the entire frequency range (dashed and dotted curves in Figure 2.11b) at the expense of a major reduction in near-field gain performance (dashed and dotted curves in Figure 2.11a) and directivity of the array at lower frequencies (Figure 2.12).

2.3.2 Summary

The maximum near-field gain design would seem to be ideal for extracting speech signals close to the microphone array while rejecting far-field

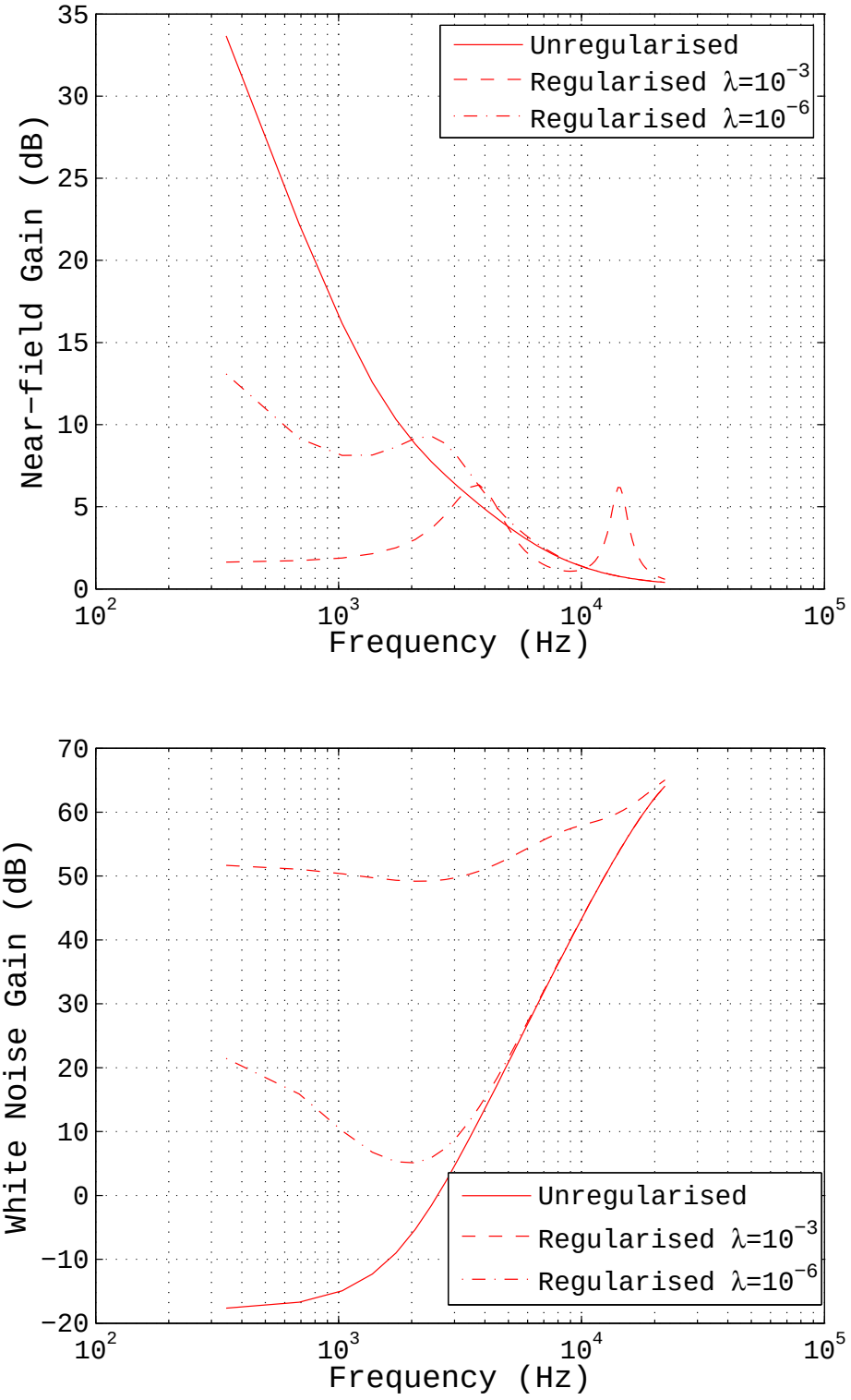


Figure 2.11: Near-field gain and white noise gain for the maximum eigenvalue beamformer and two regularised cases. The unregularised solution produces excellent near-field gain at the expense of very poor white noise gain. The regularised cases reduce near-field gain but improve white noise gain, allowing the array to handle microphone error.

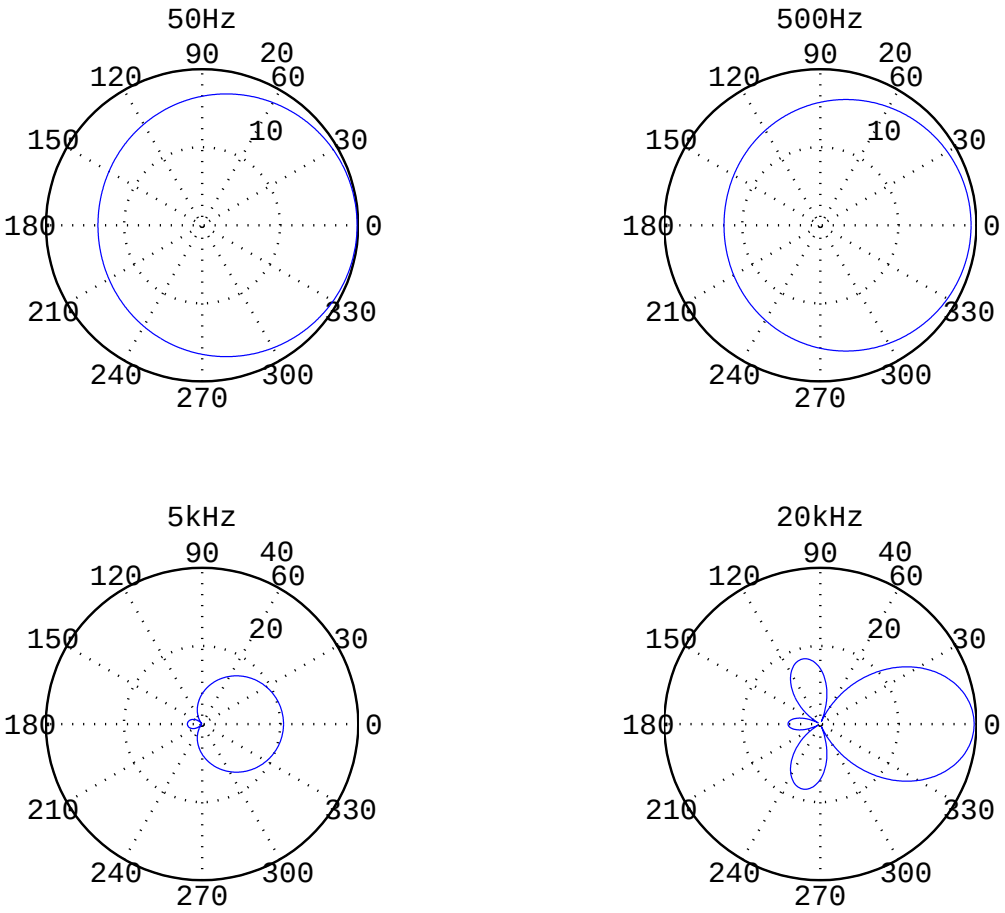


Figure 2.12: Beampatterns for the regularised maximum eigenvalue beamformer. At low frequencies, the response is similar to an omni-directional design, with little attenuation of sound sources opposite the talker. At high frequencies, the array response becomes more directional, improving attenuation of sources away from the talker.

interference. However, poor white noise gain (like the second-order differential array) prevents this type of array from achieving its potential in a real-world scenario. This technique relies on the assumption of the desired signal originating from a very specific and localised point — an unrealistic assumption.

The concept of maximising the near to far signal ratio is a useful design criteria, as it essentially defines a signal to noise ratio optimisation problem. A more robust multiple point extension of the near-field gain technique is developed in Section 2.5 to handle the deficiencies of the single point near-field maximisation design.

2.4 Iterative Method for Specifying White Noise Gain

In the near-field gain optimisation design (Section 2.3), the output of the array due to a near-field source was maximised relative to the average far-field power. The far-field power was represented using a far-field correlation matrix, obtained by averaging over far-field sources in 3D. By switching this matrix to the identity matrix, the equations now solve the white noise gain maximisation problem. In some applications it may be desirable to specify a white noise gain (or minimum white noise gain) to ensure robustness of the array. By inserting a regularisation matrix into R_{n}, the white noise gain of the array changes (along with the near-field gain and beampattern). However, deriving the required regularisation parameter to obtain the desired white noise gain is not a simple mathematical procedure [17], forcing the use of iterative techniques.

An iterative technique would involve searching a range of regularisation parameters, λ, to find a solution for the weights with the desired white noise gain. Modifying equation 2.39,

$$\mu_{opt} = \frac{\boldsymbol{w}^H(R_{\mathrm{n}} + \lambda_{opt} I_M)\boldsymbol{w}}{\boldsymbol{w}^H(R_{\mathrm{f}} + \lambda_{opt} I_M)\boldsymbol{w}} \tag{2.54}$$

where μ_{opt} is the near-field gain and λ_{opt} is the regularisation parameter giving the desired white noise gain. First the bounds on the range of possible white noise gain values must be established. The white noise gain value could be computed for a set of regularisation matrices with λ values ranging from 10^{-12} to 1. The simulations showed (Figure 2.13) that the white noise gain monotonically increased as the regularisation parameter increased to some limit, at which point increasing the regularisation had no effect — justifying the upper limit of regularisation.

The algorithm developed was a simple binary search of the 'λ space' to find the appropriate regularisation value that produced the desired white noise gain value (within some tolerance ϵ).

In Figure 2.13 the effects of regularisation on a maximal near-field design are seen for an array with a source-array centre distance of 4cm. Here 13 values of λ were selected from 10^{-12} to 1. In general, below 5kHz, the white noise gain decreases with decreasing lambda values. However, it is apparent that constraining the white noise gain has adverse effects on the near-field gain of the array.

A simple simulation was prepared to test the algorithms ability to constrain the white noise gain. Three white noise values were tested, -10dB, 0dB and 10dB. The algorithm was run using these parameters and the near-field gain and white noise gain for a set of frequencies was calculated.

The effect of optimising for white noise gain on the near-field gain performance is demonstrated in Figures 2.14a and 2.14b. Constraining the white noise gain to be above 0dB (a desirable property) shows negative effects on the near-field gain performance of the array. This demonstrates that careful use of constraining the white noise gain is required in order to prevent the array from loosing its near-field gain (and corresponding far-field noise reducing) properties.

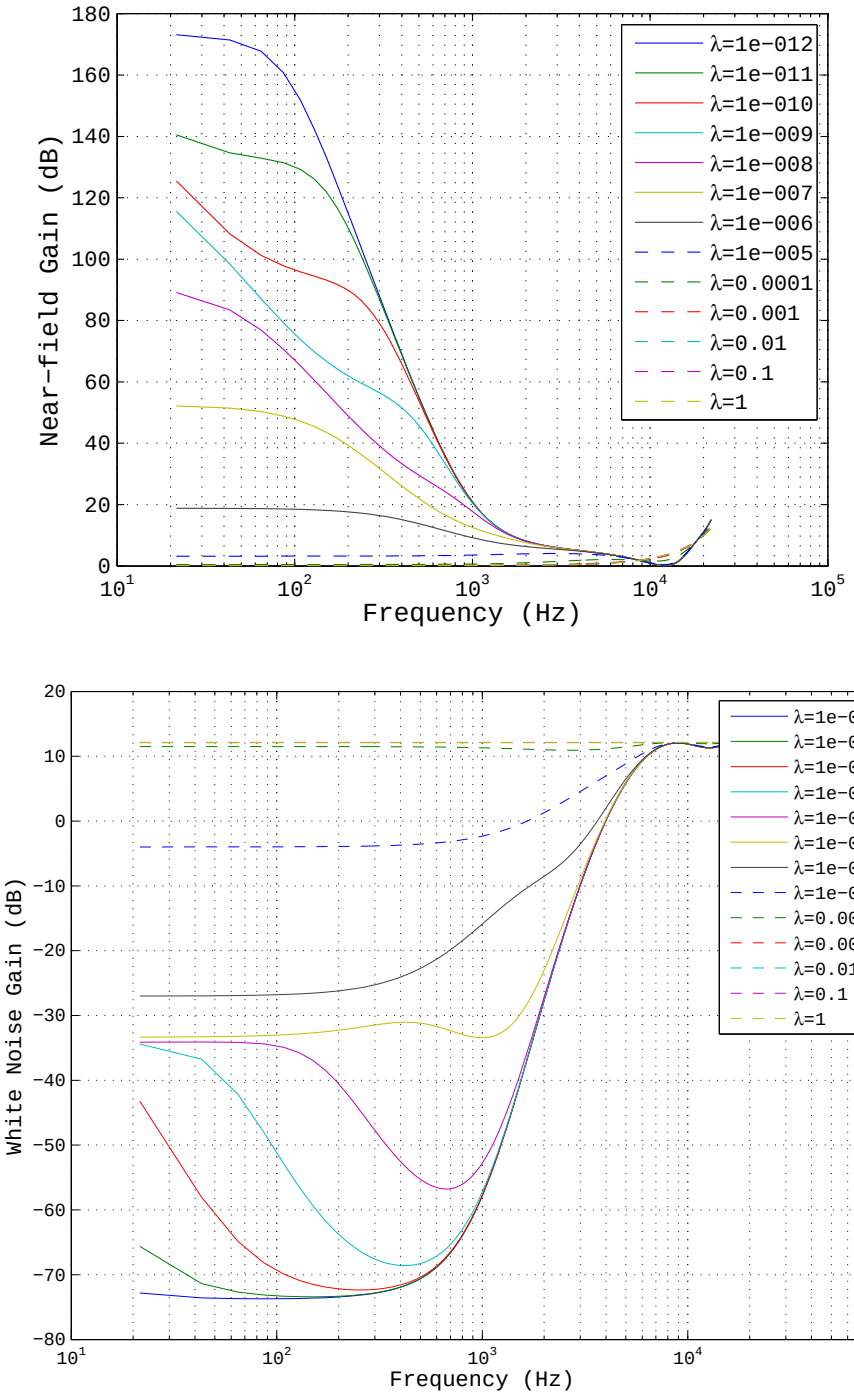


Figure 2.13: 2.13a Near-field gain for the maximum near-field gain design with varying regularisation and 2.13b, the white noise gain of the maximum near-field design using varying regularisation. In general poor near-field gain is associated with good white noise gain.

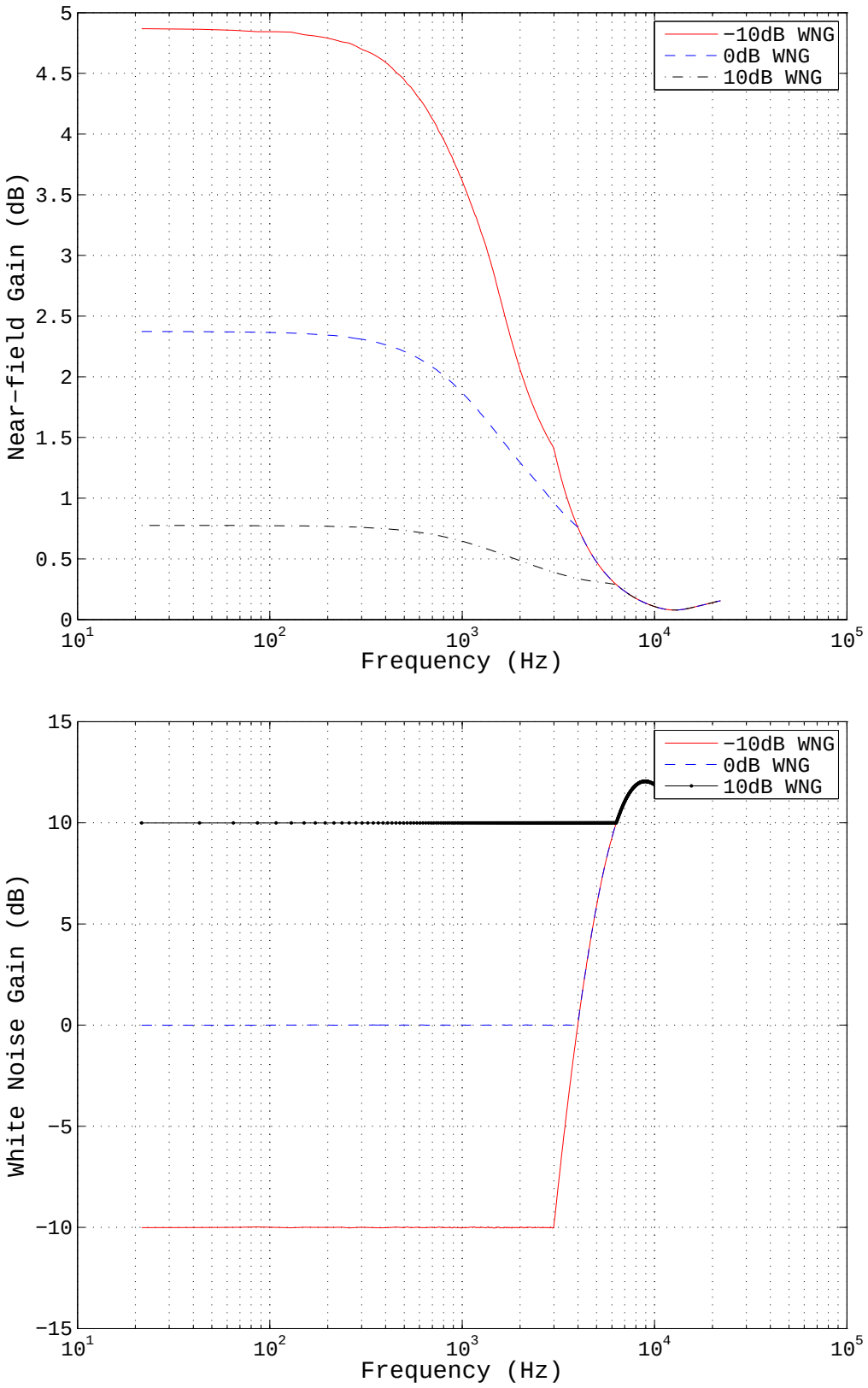


Figure 2.14: Near-field gain and white noise gain for a regularised maximum near-field gain design with a target white noise gain of -10dB, 0dB and 10dB. Note the effects of improving white noise gain on near-field gain.

2.5 Near-field Least Squares Beamforming

In this section, three beamforming designs based on least squares solutions [22][23][24] for the weights required to produce a pre-determined response are presented and evaluated.

The objective of a least squares weight solution is to find the array weight vector $\boldsymbol{w}$ which produces a desired response at a point $\boldsymbol{x}$ (or as an extension, a set of points).

The output of a near-field microphone array due to a source at location $\boldsymbol{x}$ is given as

$$s(k, \boldsymbol{x}) = \sum_{m=0}^{M-1} w_m \frac{e^{-ik\|\boldsymbol{x}-\boldsymbol{x_m}\|}}{\|\boldsymbol{x}-\boldsymbol{x_m}\|} \tag{2.55}$$

where $\boldsymbol{x_m}$ denotes the location of the m^{th} microphone.

Consider the inverse problem to microphone beamforming of specifying a set of pressure values at multiple locations (the loudspeaker beamforming problem). First defining a collection of points at which a specified pressure is to be controlled,

$$X = [\boldsymbol{x_1 x_2} \ldots \boldsymbol{x_N}] \tag{2.56}$$

where each vector $\boldsymbol{x}_n$ represents the position of a point in the set.

The pressure at each of the points in X can be obtained by calculating the product of the transfer functions and the microphone weights, as in equation 2.55. Since there is now a vector of pressure values, a matrix form can be used.

$$\boldsymbol{p}(k, X) = H\boldsymbol{w} \tag{2.57}$$

$$H_{nm} = \frac{e^{-ik\|\boldsymbol{x}_n-\boldsymbol{x}_m\|}}{\|\boldsymbol{x}_n-\boldsymbol{x}_m\|} \tag{2.58}$$

where $\boldsymbol{p}$ is a vector describing the pressure for the set of points in X; H is a matrix containing the transfer functions from each point to each

microphone in the array, and $\boldsymbol{w}$ the weights required to produce the desired $\boldsymbol{p}$ vector.

In the loudspeaker problem, the pressure vector $\boldsymbol{p}$ is known and the objective is to design beamforming weights which produces this output. In the microphone array case, instead of producing a desired pressure vector, the objective is to capture signals originating from this region. In either case, the weights can be solved using the standard least squares solution

$$\boldsymbol{w} = (H^H H)^{-1} H^H \boldsymbol{p} \tag{2.59}$$

(or to improve robustness, by including an additional Tikhonov regularisation [25] term — λI_M)

$$\boldsymbol{w} = (H^H H + \lambda I_M)^{-1} H^H \boldsymbol{p} \tag{2.60}$$

where λ is a small value — 10^{-3} in the simulations developed later.

To adapt this to a microphone array, the equations remain the same, however the pressure vector is unknown — the signal originating from the points in the matrix X are not known in advance. Instead, the pressure vector represents hypothetical point sources which may exist in the region of interest. These point sources may be weighted in order to produce a solution which minimises a signal in a region where the talker is not expected to be located for example.

$$\boldsymbol{p} = [p_1 p_1 p_2 ... p_N]^T$$
$$p_n = A_n \frac{e^{-ikr_n}}{r_n}$$

2.5.1 Simulations

Simulations were conducted using four microphones arranged in a circular array with a diameter of 7.8mm. Three least squares designs were

evaluated for near-field gain and white noise gain performance.

2.5.2 Simple Least Squares Design

The first least squares design was one in which a simple desired response was specified. The desired response was a simple design to accept sources located in a 90° wide arc centred on the source at 0°, located 4cm from the centre of the array.

$$\boldsymbol{s}(k, r = r_0, \boldsymbol{\theta}) = \begin{cases} \frac{e^{-ikr}}{r} & \text{if } \theta \in [\frac{-\pi}{4}, \frac{\pi}{4}) \\ 0 & \text{otherwise} \end{cases} \tag{2.61}$$

A regularisation parameter of 10^{-3} was used with this beamformer.

The ideal response was to reject sound arriving in the complementary region. The least squares solution does not achieve this due to array constraints — too few degrees of freedom, dictated by the number of microphones, to create enough nulls to approximate the pattern. However the solution does significantly attenuate near-field sources opposite the talker (as seen in Figure 2.15).

The simple design set out to retain signals originating in a 90 degree region centred on the talker whilst attenuating signals elsewhere. The solution of this least squares problem delivered a beamformer which approximated these requirements reasonably well. The beampatterns show good rejection of sources located opposite the talker as intended (Figure 2.15). Near-field gain performance is good below 10kHz — covering the speech band of the frequency spectrum. Indicating that speech sources located near the array would be amplified relative to far interferers. White noise gain is controlled reasonably well throughout the frequency range tested, however shows poorer performance at low fre-

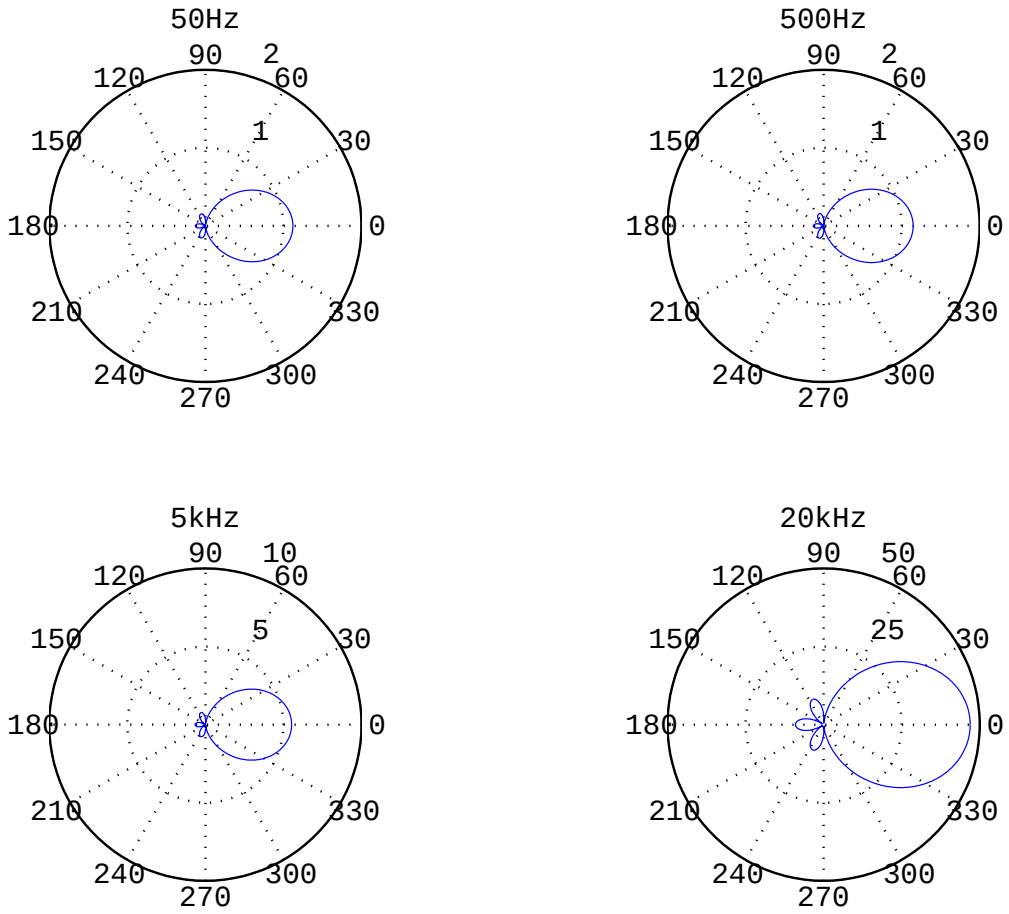


Figure 2.15: Beampatterns for the simple LSQ beamformer. The least squares solution for the pattern specified in (2.61) is a reasonable match, showing good rejection of sources opposite the talker.

quencies, where the design has greater difficulty producing a highly directional response, a consequence of the small array size.

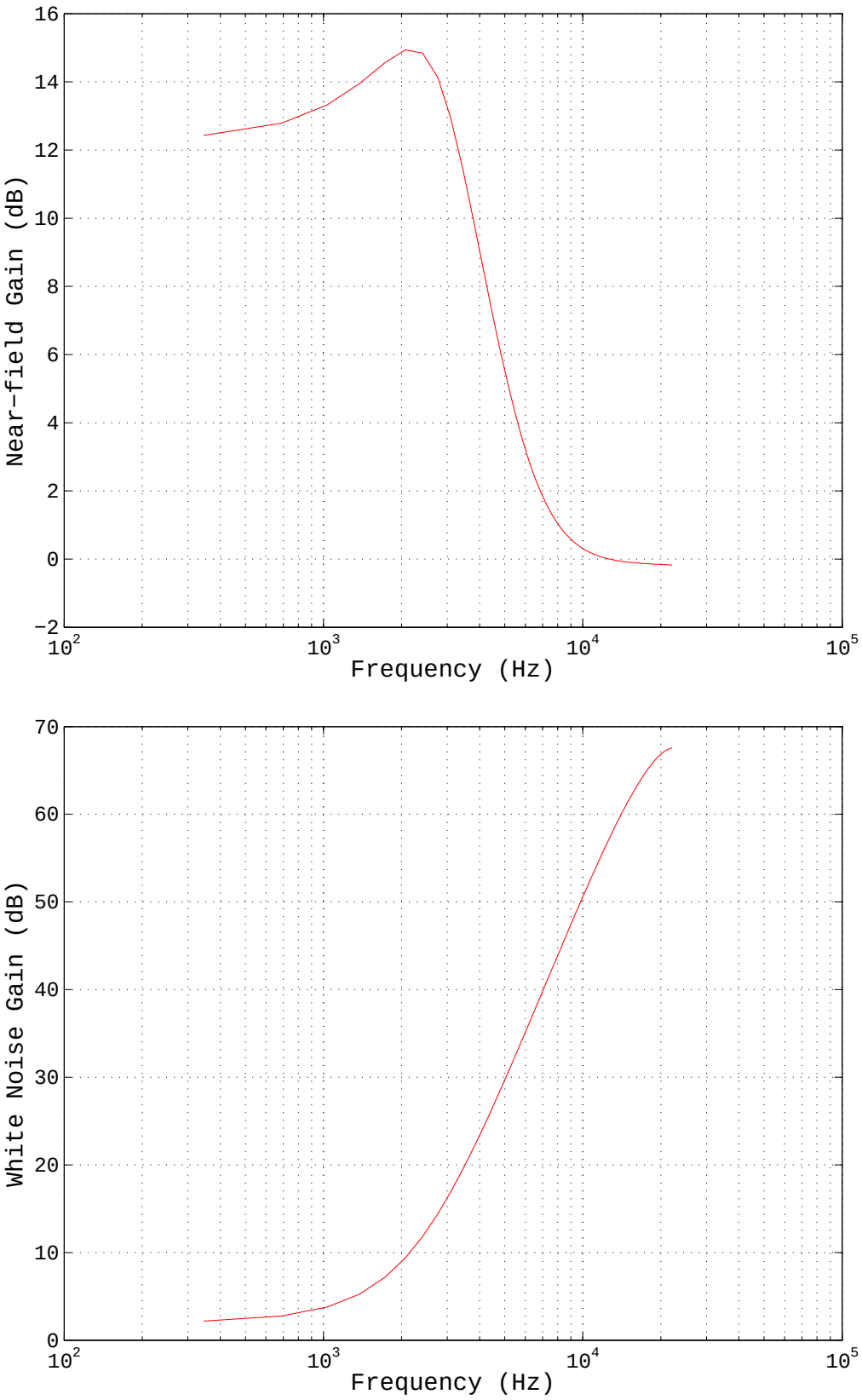


Figure 2.16: Near-field gain (2.16a) and white noise gain (2.16b) for the simple LSQ beamformer. Near-field gain is good throughout the frequency speech range of speech (<4kHz). White noise gain is acceptable at low frequencies, improving at higher frequencies.

2.5.3 'Wedge' Least Squares Design

The second least squares design extended the design region from a circle at 4cm from the centre of the array to concentric circles about the origin in order to improve robustness — effectively a wedge pattern. The goal was to allow the talker to move slightly closer or further away from the array without compromising the performance of the beamformer.

$$s(k, \boldsymbol{r}, \boldsymbol{\theta}) = r \in [r_1 r_2 ... r_N] \begin{cases} \frac{e^{-ikr}}{r} & \text{if } \theta \in [\frac{-\pi}{4}, \frac{\pi}{4}) \\ 0 & \text{otherwise} \end{cases} \tag{2.62}$$

The wedge design shows slightly reduced near-field gain compared with the simple least squares design. This is a side effect of extending the area over which to beamform. The simple design used a specified beampattern at a specific distance (the expected location of the talker). The wedge design extended the range of distances to model more realistic conditions (the talker moving backwards and forwards from/to the array).

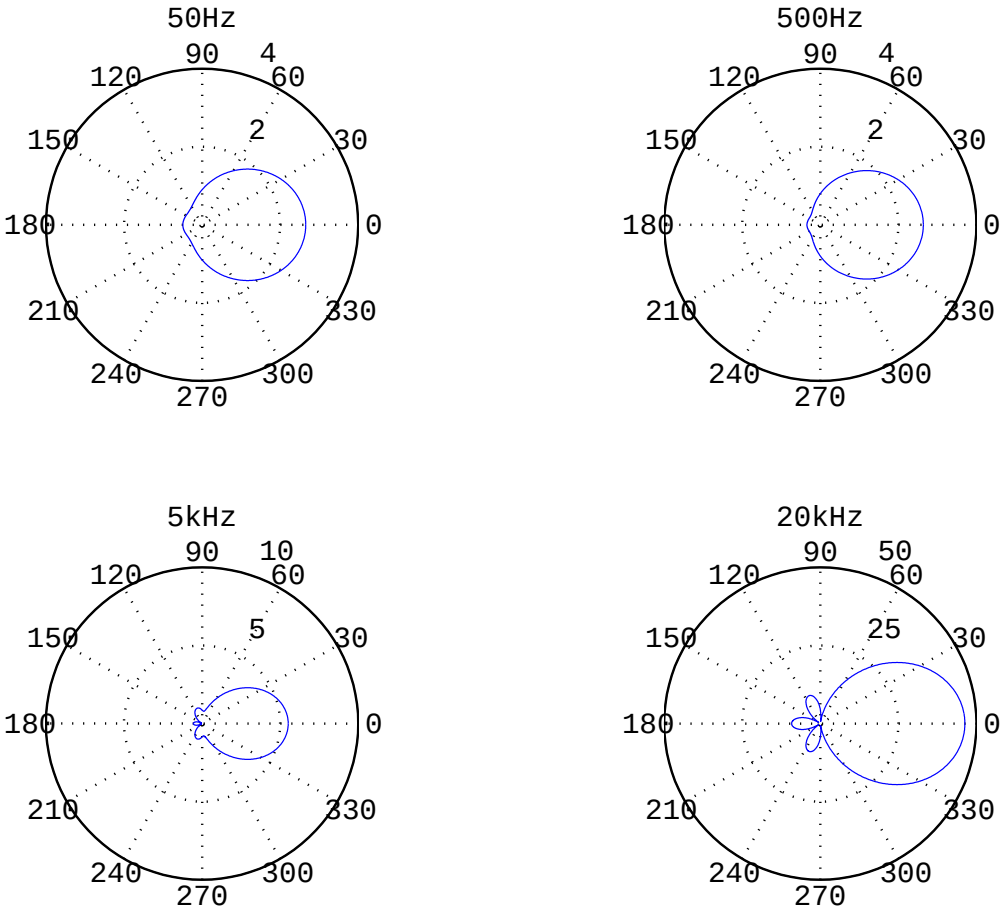
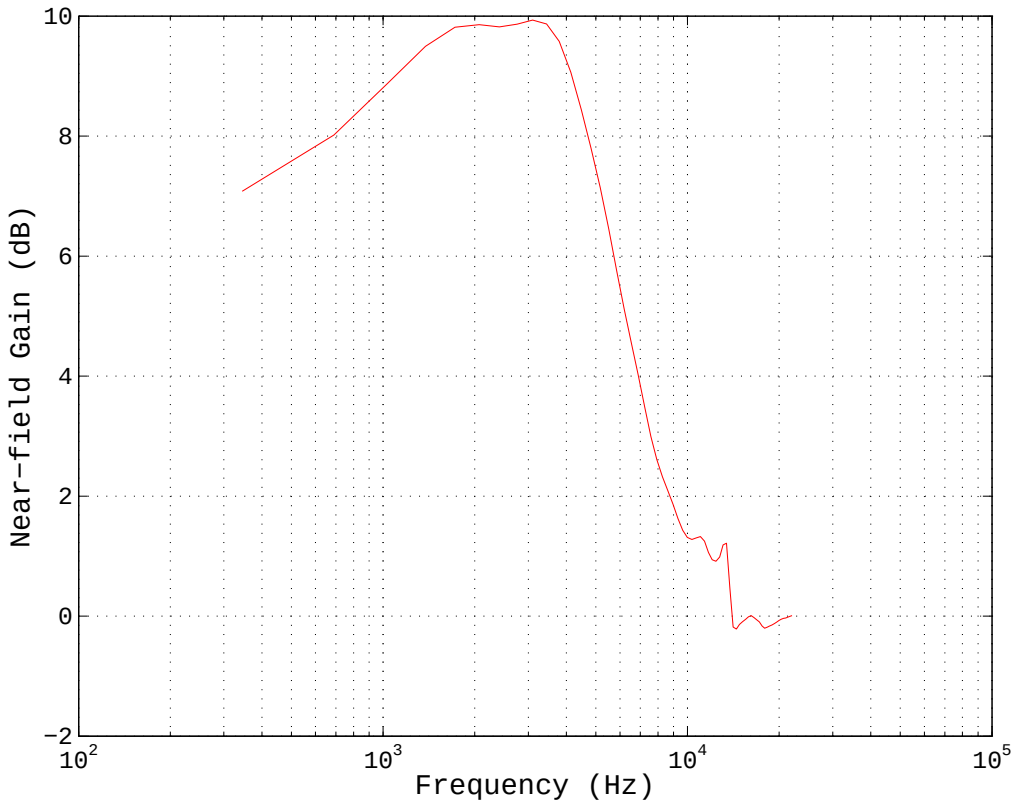


Figure 2.17: Beampatterns for the wedge LSQ beamformer. The patterns differ slightly from the simple LSQ design as a result of forcing more constraints in distance.



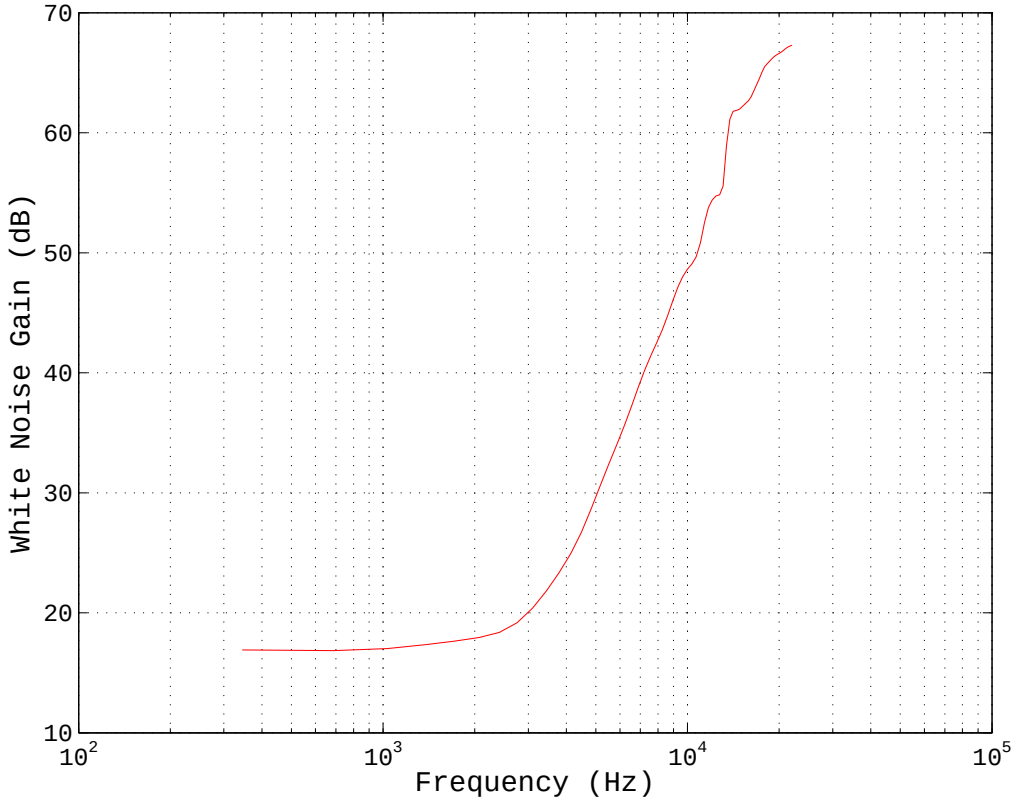


Figure 2.18: Near-field gain (2.18a) and white noise gain (2.18b) for the wedge LSQ beamformer. Near-field gain is slightly worse than the simple LSQ design. However, white noise gain is improved.

2.5.4 Maximum Ratio Least Squares Design

The third technique involved creating two regions near the array — one where the talker would be expected to be located and the other directly opposite. The least squares design attempts to maximise the signal received from the talker region relative to the symmetric shadow region. The two regions were defined as circles with radii of 2cm located 4cm from the centre of the microphone array.

A least squares design for this involves designing two regions with transfer functions given by two matrices H_1 (the region near the talker) and H_2 (the region opposite), and two desired response vectors $\boldsymbol{p}_1$ — a vector of responses corresponding to point sources originating from the talker region, and $\boldsymbol{p}_2$ — a vector of zeros. Combining the matrices into a single transfer function H and the desired response vectors into a single vector $\boldsymbol{p}$, a least squares solution can be obtained for the beamforming weights using the standard technique (2.60).

Equivalently, this least squares design can be transformed into a maximum eigenvalue design (such as the near-field gain maximisation design presented in Section 2.3) by recognising that the objective is to effectively maximise the ratio of the pressure in the talker region to the pressure in the opposing null region.

The least squares beamforming problem can be converted into the maximum eigenvalue problem by defining equivalent near-field and far-field correlation matrices. Earlier, two transfer function matrices were defined representing the response from points in two defined regions (one at/near the talker, the other on the opposite side of the microphone array).

The near-field correlation matrix becomes,

$$R_1 = H_1^H H_1 \tag{2.63}$$

a correlation matrix representing the correlation between the set of points near the talker and the microphones.

The 'far-field' correlation matrix becomes,

$$R_2 = H_2^H H_2 \tag{2.64}$$

a correlation matrix representing the opposing region.

The solution for the weights is the same generalised eigenvalue problem as before.

$$\mu = \frac{\boldsymbol{w}^H R_1 \boldsymbol{w}}{\boldsymbol{w}^H R_2 \boldsymbol{w}} \tag{2.65}$$

Provided $R_\mathrm{f}^H R_\mathrm{f}$ is invertible, the weights can be obtained from the eigenvector corresponding to the largest eigenvalue of the matrix

$$C = (R_2^H R_2)^{-1} R_1^H R_1 \tag{2.66}$$

At low frequencies, the matrices R_n and R_f may be ill-conditioned, corresponding to poor white noise gain, requiring the use of Tikhonov regularisation to solve for the eigenvectors.

$$C' = (R_2^H R_2 + \lambda I_M)^{-1} (R_1^H R_1) \tag{2.67}$$

where lambda is some small regularisation parameter set to 10^{-3} in the simulations.

This technique is similar to the maximum near-field gain design with the exception of using multiple source points for the near-field correlation matrix and replacing the far-field correlation matrix with a shadow near-field correlation matrix.

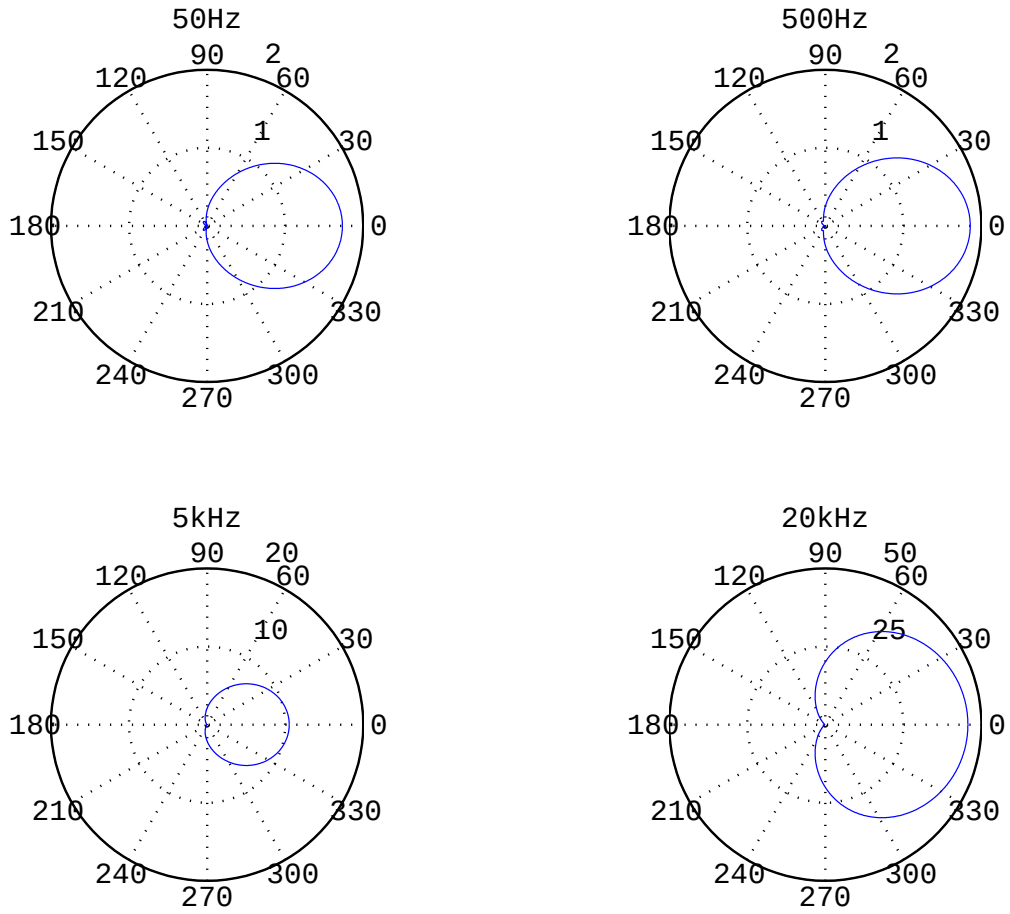


Figure 2.19: Beampatterns for the maximum ratio LSQ beamformer. This design produces greater attenuation of sources opposite the talker than the simple and wedge LSQ designs.

The ratio technique showed good overall performance. Near-field gain was significant ($\geq$ 10dB) for the speech band of the spectrum, in contrast with the other array designs tested — most of which declined above 1kHz. White noise gain is typically lower than most of the other designs evaluated, however it lies above 10dB for the entire frequency range tested, indicating no significant problems with implementation. Additional white noise gain could be achieved by increasing the regularisation parameter (as is the case for the other least squares designs).

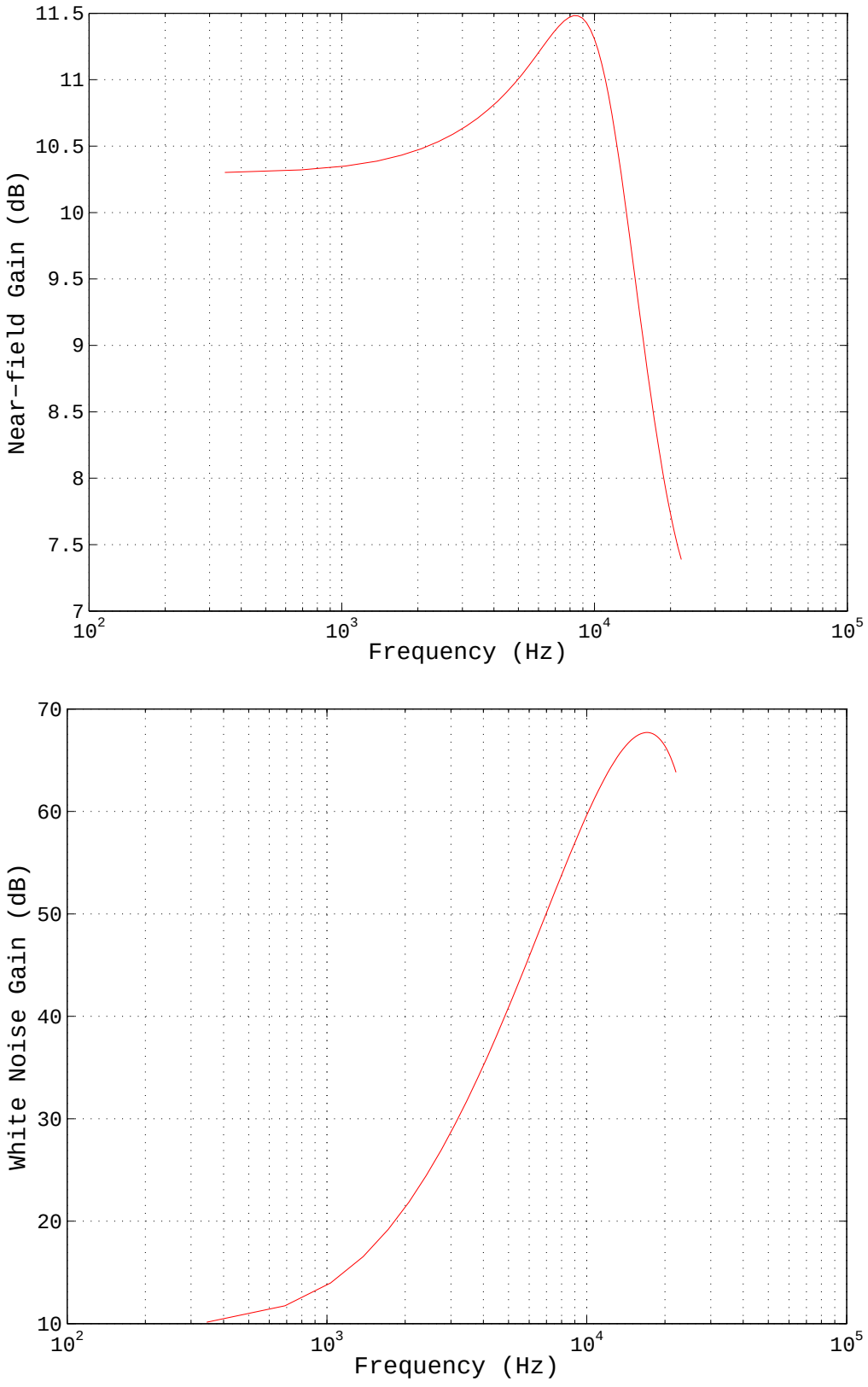


Figure 2.20: Near-field gain (2.20a) and white noise gain (2.20b) for the maximum ratio LSQ beamformer. Near-field gain is good throughout the entire frequency range, with less variation than the other two LSQ designs. White noise gain is better than the simple LSQ design, but slightly worse compared to the wedge design.

2.5.5 Least Squares Summary

Of the three least squares methods described, the ratio technique in which the signals originating from one area are maximised relative to a secondary region shows good overall performance. Near-field gain is relatively flat across the frequency range compared to the simple and wedge designs and white noise gain is competitive with the two other designs.

The advantage of least squares methods over differential arrays and the single-point near-field gain optimised design is balanced performance between near-field gain and white noise gain robustness.

2.6 Conclusion

Several beamforming designs were evaluated for their suitability for isolating speech signals from background noise. Two main performance criteria were tested: near-field gain, the measure of near signal isolation from distant interferers (Figure 2.21); and white noise gain, the measure of array robustness to errors in microphone placement and mismatch in the microphone responses (Figure 2.22).

Second order differential arrays show excellent near-field gain for much of the frequency spectrum where speech is concentrated, exceeding the performance of the other arrays tested below 1kHz. However, poor white noise gain indicates that the array would have difficulty handling minor imperfections in the set up, requiring perfect microphone matching and talker placement to ensure good performance.

The ratio least squares technique shows good near-field gain throughout the entire frequency range tested (up to 22kHz). Near-field gain performance exceeds that of the other designs above 4kHz, while maintaining good relative performance below this point. White noise gain is good throughout the entire frequency range, suggesting that the array is capable of tolerating imperfections in microphone responses or a slight perturbation in the talker location.

The simultaneous requirements of good near-field gain and white noise gain suggests that the ratio least squares technique is ideal as a beamformer/nullformer in a speech capture system.

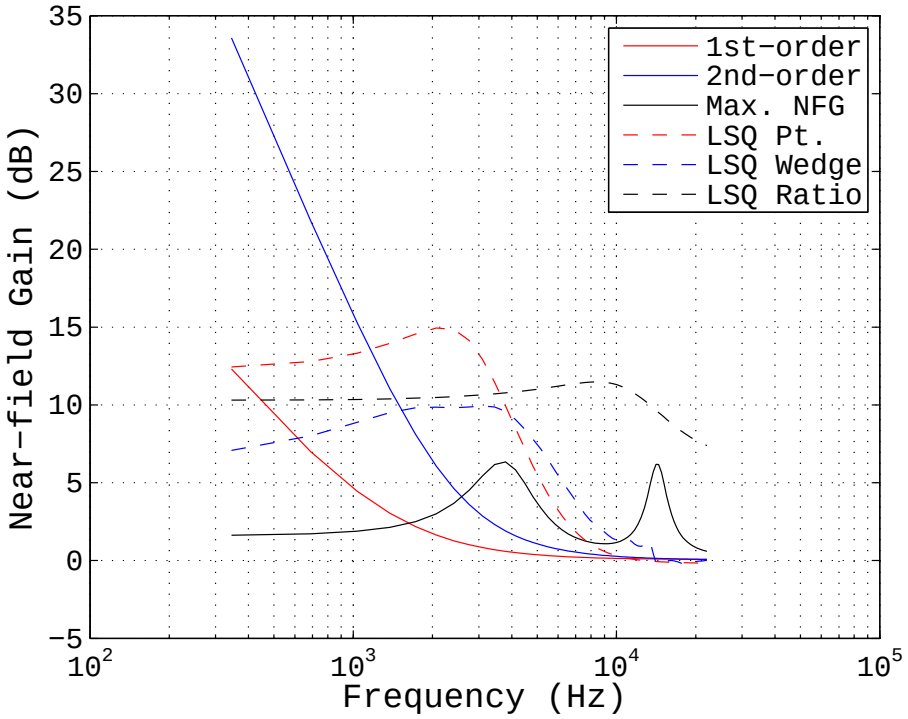


Figure 2.21: Near-field gain for each of the array types. The ratio LSQ technique presented in Section 2.5 stands out for its near flat near-field gain throughout the frequency range.

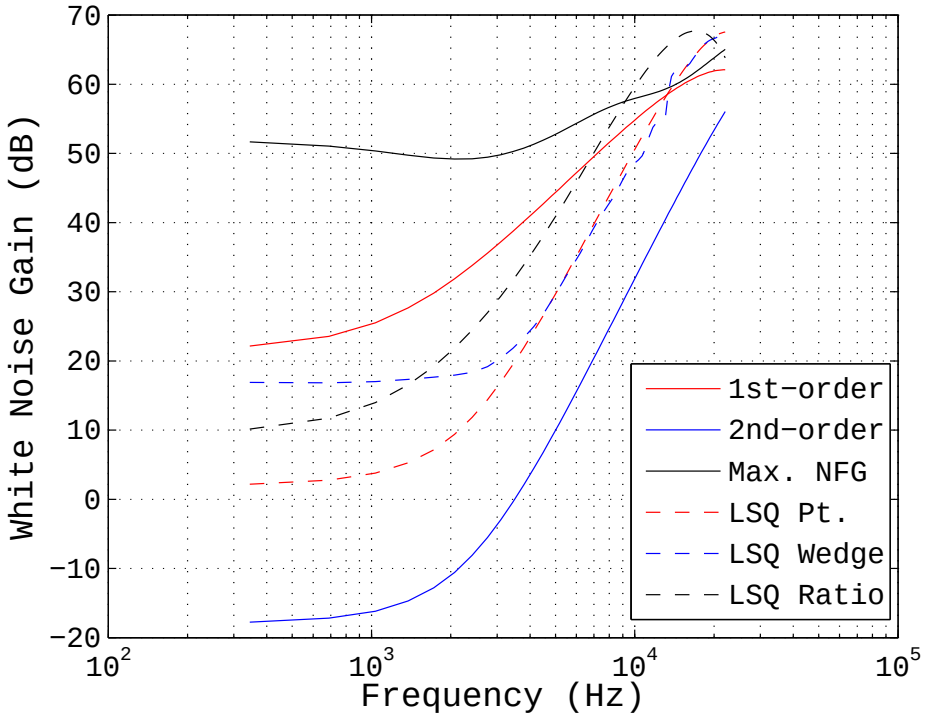


Figure 2.22: White noise gain for each of the array types. The second order differential array stands out as a poor performer.

Chapter 3

Adaptive Filters and Noise Cancellation

In this chapter, adaptive filtering is introduced. A brief overview of the various applications of adaptive filtering is provided before introducing the main application in this thesis: noise cancellation. The least squares filter (LMS filter) is derived from a gradient descent method used in quadratic optimisation and analysed for potential noise filtering properties in the context of a simple single frequency cancellation scenario. Deficiencies in adaptive noise cancellation when handling cross-talk is discussed and a novel method for counteracting undesirable behaviour is presented. Simulations with adaptive filters are presented and discussed. Finally, more advanced adaptive filtering systems (CTRANC, SAD and RLS) are briefly discussed.

3.1 Adaptive Filters

Adaptive filters are used for a variety of applications in signal processing. Uses include system identification, linear prediction and noise cancellation [11]. In each of these cases, the adaptive filter seeks to minimise an error between a 'desired' signal and the adaptive filter output (a filtered 'reference' signal).

3.1.1 System Identification

In a system identification scenario the adaptive filter attempts to calculate an unknown system by minimising the difference between the unknown system output (with a known input) and the adaptive filter output when fed with the original known signal. Various applications of this method are highlighted in [12].

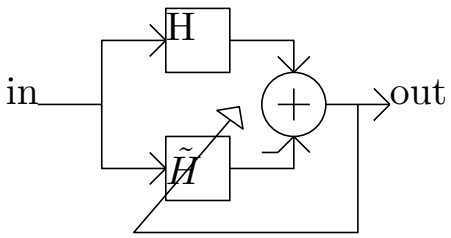


Figure 3.1: Simple system identification process

3.1.2 System Equalisation

This application of adaptive filtering seeks to find a filter which inverts (or equalises) an unknown system with a known input. The original input and the filtered inputs are used to adjust the adaptive filter to minimise the error between the original signal and filtered system output.

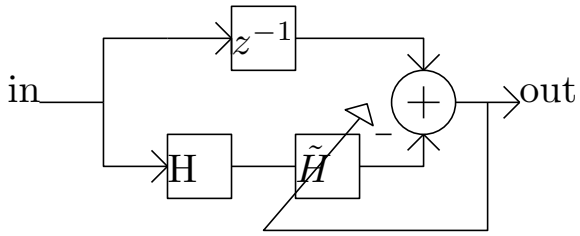


Figure 3.2: Simple equalisation process

3.1.3 Linear Prediction

Adaptive filters can be used to calculate the coefficients of a linear prediction system. Applications of this particular method include data compression and the closely related field of speech analysis/synthesis [26] [27]. The prediction coefficients are calculated by feeding the original signal (as the primary) and a delayed version of the original signal (as the reference) into the adaptive filter system (Figure 3.3).

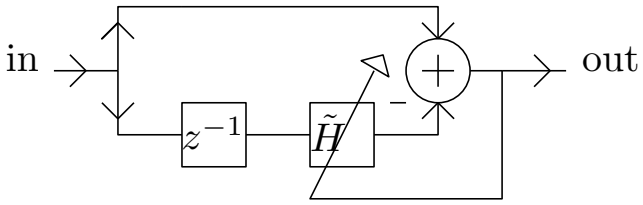


Figure 3.3: Linear predictor

3.2 Noise Cancellation

Noise cancellation is another popular application of adaptive filtering. The process involves filtering a primary (or 'desired' as known in the literature) noisy signal using a reference signal containing only noise. The noise is assumed to leak into the primary channel via some unknown process (modelled as a filter, H). The adaptive filter attempts to model the unknown filter in order to remove the noise from the primary channel, leaving the uncorrelated original signal.

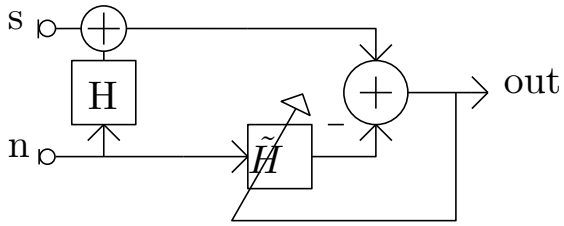


Figure 3.4: Simple noise cancelling system

In a 2-channel noise canceller two signals, a source and interferer, are received by two microphones having undergone a convolutive mixing

process. For a single frequency, this process can be expressed by the a sequence of matrix equations to follow. First a convolutive mixing matrix A is defined with elements H_{ij} describing the transfer functions between each source and receiver at a single frequency (in general these elements are Toeplitz convolution matrices describing filters, however for simplicity they are instantaneous single values here).

$$A = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

The vector (in the instantaneous case), $\boldsymbol{s}$, containing the source (s_1) and interferer signals (s_2) is,

$$\boldsymbol{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

The mixing process results in a vector $\boldsymbol{x}$ representing what is received at each microphone.

$$\boldsymbol{x} = A\boldsymbol{s} = \begin{bmatrix} H_{11}s_1 + H_{12}s_2 \\ H_{21}s_1 + H_{22}s_2 \end{bmatrix}$$

A simple noise cancelling adaptive filter system takes the output of the first ('primary') microphone and subtracts the output of the adaptive filter convolved with the second ('reference') microphone. The filter coefficients are updated using this estimate of the desired signal. In noise cancellation, the objective is to find some filter $\tilde{H}$ which minimises the unwanted interferer in the primary channel. In matrix form this can be expressed as,

$$\tilde{\boldsymbol{s}} = BA\boldsymbol{s} = \begin{bmatrix} 1 & -\tilde{H} \\ 0 & \tilde{H} \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \tag{3.1}$$

Expanding terms, the signal estimate vector $\tilde{\boldsymbol{s}}$ becomes:

$$\tilde{\boldsymbol{s}} = \begin{bmatrix} H_{11}s_1 + H_{12}s_2 - \tilde{H}(H_{21}s_1 + H_{22}s_2) \\ \tilde{H}(H_{21}s_1 + H_{22}s_2) \end{bmatrix} \tag{3.2}$$

with $\tilde{\boldsymbol{s}}$ now representing an estimate of the original source and interferer signals.

To remove completely remove noise from the primary channel, and assuming there is no cross-talk between the original signal and the reference channel, $\tilde{H}$ should converge to

$$\tilde{H} = \frac{H_{12}}{H_{22}}$$

leaving the estimated signals as

$$\tilde{\boldsymbol{s}} = \begin{bmatrix} \left(H_{11} - \frac{H_{12}H_{21}}{H_{22}}\right) s_1 \\ \frac{H_{12}H_{21}}{H_{22}}s_1 + H_{12}s_2 \end{bmatrix} \tag{3.3}$$

under the earlier assumption about source to reference cross-talk ($H_{21} = 0$), the estimated signals are therefore,

$$\tilde{\boldsymbol{s}} = \begin{bmatrix} H_{11}s_1 \\ H_{12}s_2 \end{bmatrix} \tag{3.4}$$

Now consider source-to-reference crosstalk ($H_{21} \neq 0$).

$$\tilde{\boldsymbol{s}} = \begin{bmatrix} \left(H_{11} - \frac{H_{12}H_{21}}{H_{22}}\right) s_1 \\ \frac{H_{12}H_{21}}{H_{22}}s_1 + H_{12}s_2 \end{bmatrix} \tag{3.5}$$

The ideal filter $\tilde{H}$ removes the interferer from the primary channel, however, the resulting signal no longer perfectly matches the original signal. The signal is now attenuated by a factor relating to the level of cross-talk, which if the two microphones are very close together, presents a problem when using adaptive filters in a cross-talk environment. A novel method for compensating for cross-talk with an least mean squares filter is presented in Section 3.4 .

3.3 Least Mean Squares Filter

The least mean squares (LMS) filter is an example of a simple low complexity adaptive filter algorithm. A brief derivation is shown here.

In the previous section, the ideal filter $\tilde{H}$ represented a simple instantaneous single frequency filter to remove the interferer s_2 from the primary input x_1 to produce a resulting signal $\tilde{s}_1$ as close as possible to the original signal. Extending this to a non-instantaneous time domain scenario, the objective is to find the ideal filter in terms of a finite length filter denoted as $\boldsymbol{w}$ (for consistency with the existing literature), which reduces the error between the instantaneous primary input $x_1[n] = s_1 + \boldsymbol{w}_{\text{ct}}{}^T \boldsymbol{s}_2$ (with $\boldsymbol{w}_{\text{ct}}$ describing the interferer to primary crosstalk) and the reference input $\boldsymbol{x}_2[n] = \boldsymbol{s}_2$[n]. For simplicity, it is assumed that the filters describing the direct paths from the desired signal to the primary microphone and the noise signal to the reference are irrelevant (H_{11} and $H_{22} = 1$).

From the block diagram (Figure 3.4), the error at time index n can be expressed as

$$e(n) = x_1(n) - \boldsymbol{w}_n^T \boldsymbol{x}_2 \tag{3.6}$$

Formulating the cost function ξ as a least squares problem,

$$\xi = e^2(n) = \|x_1(n) - \boldsymbol{w}_n^T \boldsymbol{x}_2\|^2 \tag{3.7}$$

Noting that the cost function is quadratic, the optimal solution for $\boldsymbol{w}$ occurs at the point where the gradient of the cost function reaches zero. A well known algorithm for finding the optimal solution for this class of problems is the gradient descent algorithm [28]. The filter weights are adjusted at each iteration based on the current gradient of the cost function. Speed and stability of convergence is controlled by the convergence

parameter μ.

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n - \mu \bigtriangledown \xi \tag{3.8}$$

Let ξ be the estimate of the least squares error of the system, x_1 the current primary signal sample, $\boldsymbol{x}_2$ a vector containing the current and $L-1$ previous reference inputs and $\boldsymbol{w}_n$ be the current adaptive filter weights (an L-length filter).

$$\xi = E\{e^2\} = E\{(x_1 - \boldsymbol{x}_2^H \boldsymbol{w}_n)^2\} = E\{x_1^2\} + E\{\boldsymbol{w}_n^H \boldsymbol{x}_2 \boldsymbol{x_2}^H \boldsymbol{w}_n\} - 2E\{x_1 \boldsymbol{w}_n^H \boldsymbol{x}_2\} \tag{3.9}$$

Noting that $R_2 = E\{\boldsymbol{x}_2 \boldsymbol{x}_2^H\}$ is the autocorrelation matrix of the reference input and that $\boldsymbol{p} = E\{x_1 \boldsymbol{x}_2\}$ is the correlation between the instantaneous primary input and the reference, the cost function can be expressed as,

$$\xi = x_1^2 + \boldsymbol{w}_n^H R_2 \boldsymbol{w}_n - 2\boldsymbol{w}_n^H \boldsymbol{p} \tag{3.10}$$

Now taking the gradient of this function with respect to $\boldsymbol{w}_n^H$,

$$\bigtriangledown \xi = 2R_2 \boldsymbol{w}_n - 2\boldsymbol{p} \tag{3.11}$$

Factorising,

$$\bigtriangledown \xi = -2e\boldsymbol{x}_2 \tag{3.12}$$

Hence, the filter update process is as follows,

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + 2\mu e \boldsymbol{x}_2 \tag{3.13}$$

3.3.1 NLMS

A simple modification to LMS is the normalised LMS adaptive filter (NLMS). This method adjusts the convergence parameter μ based on an estimate of signal energy to stabilise the filter. The filter update equation (3.13) is normalised by the dot product of the reference vector, with

an additional small parameter alpha (set to 10^{-4} in the simulations presented in Section 3.4) to prevent infinite step sizes resulting from silence in the reference channel.

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + 2\mu e \frac{\boldsymbol{x}_2}{\boldsymbol{x}_2^T \boldsymbol{x}_2 + \alpha} \tag{3.14}$$

3.4 Power Inversion Resistant LMS

This section describes a technique for automating the shutoff for an adaptive filter operating in a high cross-talk environment. In high signal to noise ratios, adaptive filters can produce outputs which have worse characteristics than the inputs if there is significant cross-talk between the talker and the reference microphone (3.5). This poses an implementation problem. If the signal to noise ratio rises too far, the adaptive filter becomes useless in high cross-talk cases.

The objective is to design a system which prevents the power inversion problem of the standard adaptive filter noise canceller. One simple way of achieving this is to shut off the adaptive filter entirely. This will allow unwanted noise to pass through unaffected since the power inversion problem occurs at high SNR this may not be a problem for the end user. The interferer may be too quiet to cause intelligibility issues in the output signal. Such a system could be implemented by simply including a user controlled hardware switch to power off the adaptive filter under low noise conditions for example. Ideally for the application this thesis is centred on, the system would be automated for (user) simplicity. This would rely on using known information (from the primary and reference microphones) to adjust the adaptive filter in a way which achieves this power-off effect in low noise environments whilst

continuing to function similarly (performance wise) to the normal LMS adaptive filter in high noise environments.

Starting with the Cauchy-Schwarz inequality, the orthogonality of the primary and reference inputs can be determined.

$$\|\boldsymbol{x}_1\|\|\boldsymbol{x}_2\| \geq \langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle \tag{3.15}$$

$$\|\boldsymbol{x}_1\|\|\boldsymbol{x}_2\| - \langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle \geq 0 \tag{3.16}$$

$$\alpha = \|\boldsymbol{x}_1\|\|\boldsymbol{x}_2\| - \langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle \tag{3.17}$$

α in 3.17 is zero when the primary and reference are orthogonal, giving an indicator of possible power inversion problems. A simple, crude technique for shutting off the adaptive filter in this case would be to intentionally corrupt the reference signal with a large amount of uncorrelated (with the primary and reference signals) noise. Such a technique would itself need to adapt to the similarity of the primary and reference signals to ensure it did not prevent the adaptive filter from normal operation when the SNR is low. The Cauchy-Schwarz inequality can be used to define a corruption parameter which shuts off the adaptive filter when the two inputs are highly correlated while still retaining the noise reduction properties when the two inputs are distinct.

The corruption parameter needs to be large when the two vectors are highly correlated, so the inverse of α is used,

$$\beta = \frac{1}{\alpha + \gamma} = \frac{1}{\|\boldsymbol{x}_1\|\|\boldsymbol{x}_2\| - \langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle + \gamma} \tag{3.18}$$

A small parameter, γ, is used in the denominator to prevent infinite energy noise solutions. In the simulations presented earlier, this value was 10^{-4}.

The technique differs from NLMS as follows, first the noisy reference

vector is prepared by adding scaled Gaussian noise to the original reference vector. The assumption here is that this additive noise is uncorrelated with both the original signal and original noise present in the reference channel.

$$\tilde{\boldsymbol{x_2}} = \boldsymbol{x}_2 + \beta \boldsymbol{n} \tag{3.19}$$

where $\boldsymbol{n}$ is a vector containing Gaussian noise.

The adaptive filter output is calculated (as in the regular NLMS case) and subtracted from the instantaneous primary input

$$e_T = \boldsymbol{x}_1(0) - \tilde{\boldsymbol{x}_2}^H \boldsymbol{w} \tag{3.20}$$

The additive noise vector is fed through the adaptive filter and its output calculated. This is used to remove the additive noise in the final step.

$$e_N = \beta \boldsymbol{n}^H \boldsymbol{w} \tag{3.21}$$

The adaptive filter is updated using the noisy reference vector and system output.

$$\boldsymbol{w} = \boldsymbol{w} + 2\mu e_T \frac{\tilde{\boldsymbol{x_2}}}{\|\tilde{\boldsymbol{x_2}}\|} \tag{3.22}$$

Finally, the additive noise is removed from the system output.

$$e = e_T + e_N \tag{3.23}$$

Overall this system represents a further modification to the convergence parameter of LMS. In NLMS, the filter update includes a normalisation parameter to ensure stability. This modification effectively adjusts the update process to account for high cross-talk environments where the filter would normally render an undesirable (severely attenuated) output. It fails to remove noise from the system in high signal to noise cases, however it is assumed that in these situations, the signal to noise ratio is high enough to ignore noise filtering altogether.

3.4.1 Simulations

To demonstrate the effects of crosstalk induced power inversion a number of simulations were undertaken. Three scenarios were devised: simple sinusoids in noise, music in noise and speech in noise. A mixing matrix A was defined as,

$$A = \begin{bmatrix} 1 & 0.9998z^{-2} \\ 0.67z^{-2} & 1 \end{bmatrix} \tag{3.24}$$

simulating a two sources located either side of a two element array whose elements were located 1.5cm apart (the equivalent of two samples at 44.1kHz). The attenuation values were selected to simulate the positions of the talker and interferer to be located 4cm and 10m away from the centre of the array respectively.

In these simulations, the NLMS design is compared with the modified design presented in Section 3.4.

A series of signal to noise ratios from -12dB to +12dB were tested and the output of the adaptive filtering system was compared to the input 'talker' and interferer signals via an FFT performed on the final 1024 samples of the inputs/output.

3.4.2 Example: Two Sinusoids

The first simulation involves simulating two sinusoidal signals (49.5Hz and 176.4Hz) as the 'talker' and either white noise or a music track as the interferer.

At -12dB the low frequency signal is resolved in both of the techniques, however the higher frequency signal is not easily identifiable in the NLMS simulation (Figure 3.5). NLMS also shows the power inversion problem mentioned in Section 3.3, the sinusoids are attenuated significantly from the original power. By contrast, the modified NLMS method shows no attenuation of the signal at the expense of also retaining the noise perfectly. However, this is the intended design.

At 12dB the NLMS algorithm completely fails for this simulation. The two sinusoids are no longer identifiable in the spectrum, indicating that they have been attenuated severely by the algorithm. By contrast, the modified method retains the signals almost perfectly by contrast. Noise is still present as expected.

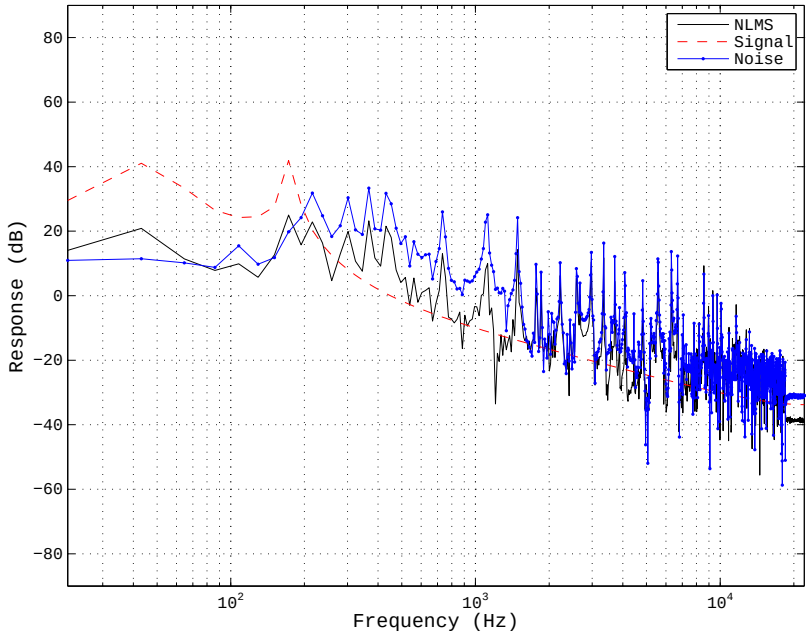


Figure 3.5: Two sine waves in music, without the modification to LMS. -12dB signal to noise ratio. The filtered signal does not resemble the original signal, closely matching the noise.

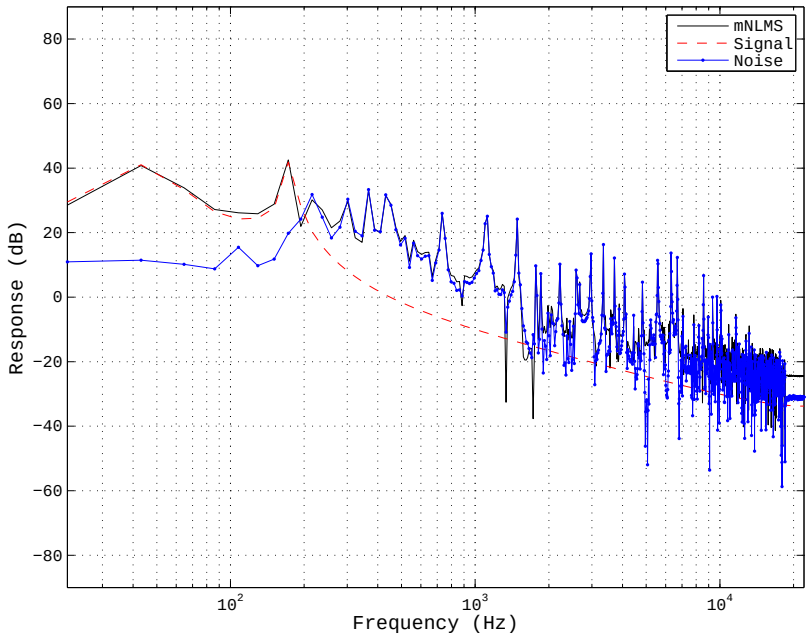


Figure 3.6: Two sine waves in music, with the modification to LMS, -12dB signal to noise ratio. The filtered signal contains both the original signal and the noise, as intended. However, at low frequencies (where the signal energy lies), an insignificant level of noise is present in the output.

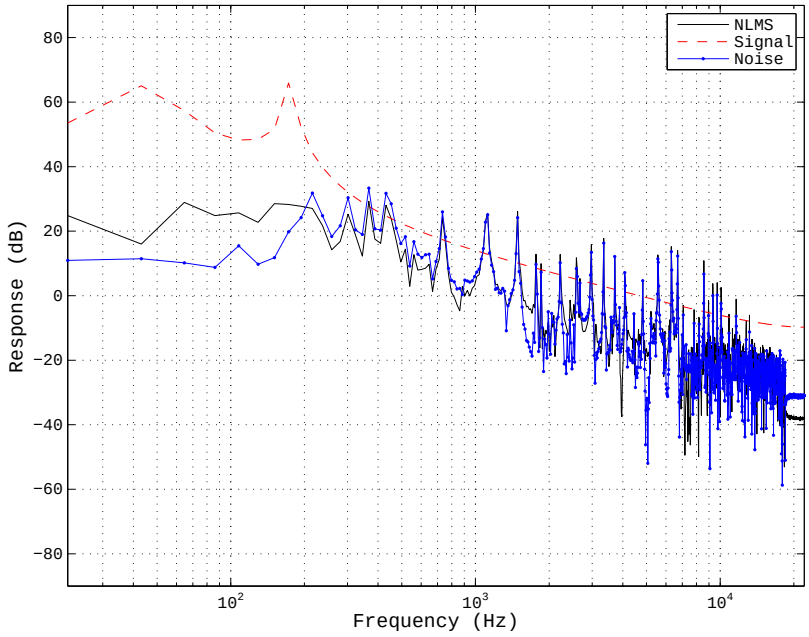


Figure 3.7: Two sine waves in music, without the modification to LMS. 12dB signal to noise ratio. NLMS fails to reconstruct the original signal due to the power inversion problem introduced earlier.

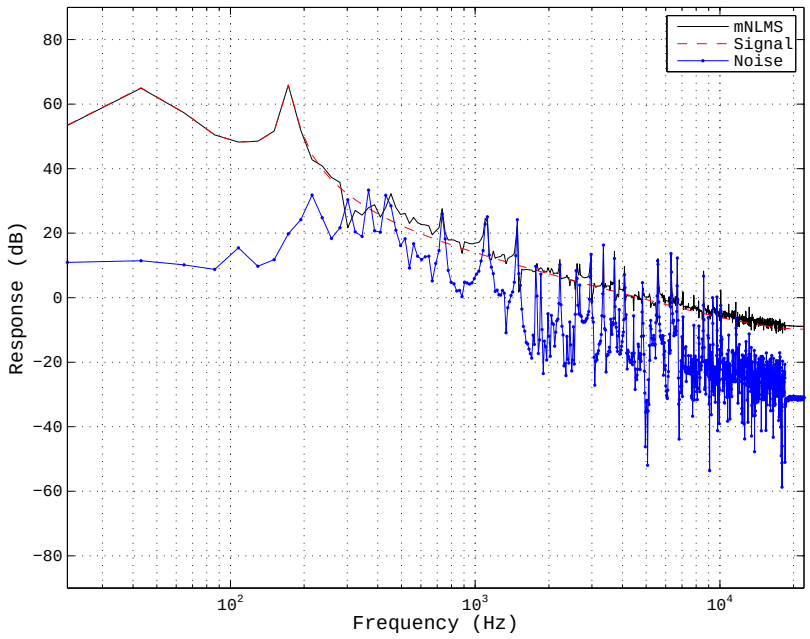


Figure 3.8: Two sine waves in music, with the modification to LMS, 12dB signal to noise ratio. The modified technique retains the original signal and some noise (at higher frequencies).

3.4.3 Example: Music

The second simulation involved using a music track as the 'talker' signal.

Similarly to the sinusoidal example, the modified NLMS method shows excellent retention of the original signal at 12dB compared with the ordinary NLMS technique. At -12dB, the NLMS method delivers a more accurate (though attenuated) spectrum of the signal compared to the modified technique. The modified technique shows worse overall performance in this simulation at -12dB, indicating that the modification does in some cases deliver worse performance than NLMS at low signal to noise ratios.

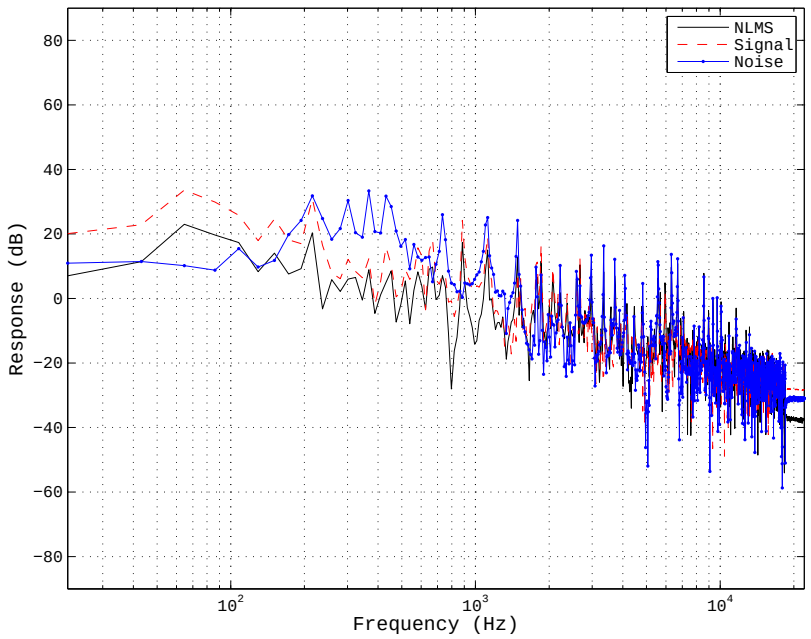


Figure 3.9: A single music track in another music track (noise), without the modification to LMS. -12dB signal to noise ratio. NLMS delivers a good result here. The signal is resolved with some attenuation.

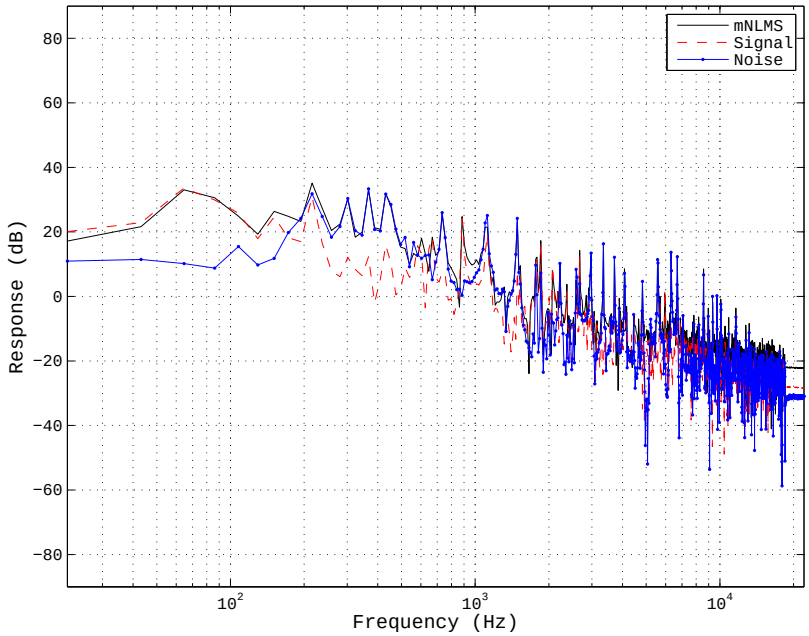


Figure 3.10: A single music track in another music track (noise), with the modification to LMS, -12dB signal to noise ratio. The modified technique retains the noise particularly at higher frequencies compared with NLMS.

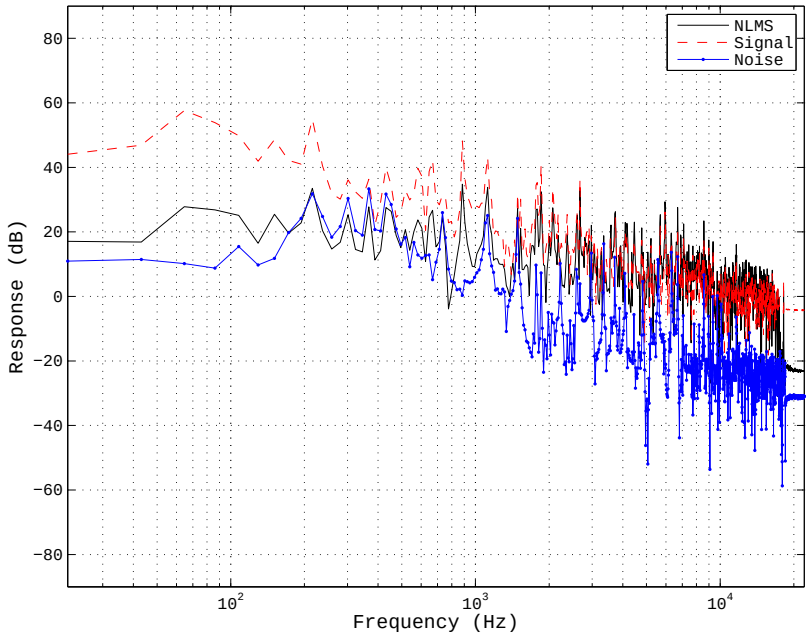


Figure 3.11: A single music track in another music track (noise), without the modification to LMS. 12dB signal to noise ratio. NLMS shows the cross-talk induced power inversion problem, and is unable to resolve the original signal.

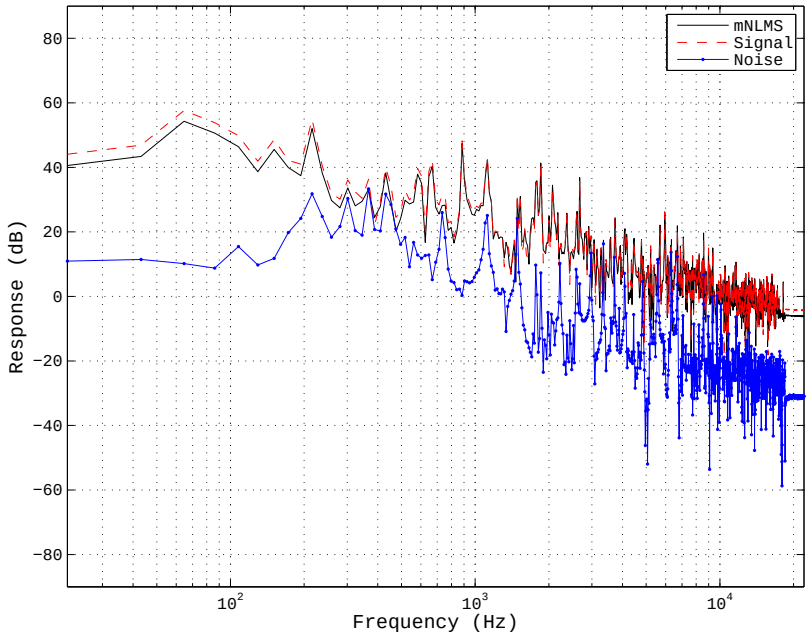


Figure 3.12: A single music track in another music track (noise), with the modification to LMS, 12dB signal to noise ratio. The modified NLMS technique shows good performance, closely matching the original signal while still retaining some noise, as expected.

3.4.4 Example: Speech

The final simulation involved using a speech signal (from the TIMIT database) as the 'talker' signal.

Performance of the two systems in the speech simulation is similar to the music simulation. At -12db SNR, the modified NLMS technique (Figure 3.14) shows slightly worse performance than the regular NLMS algorithm (Figure 3.13). At 12dB, the modified technique shows excellent retention of the original signal where the regular NLMS algorithm fails.

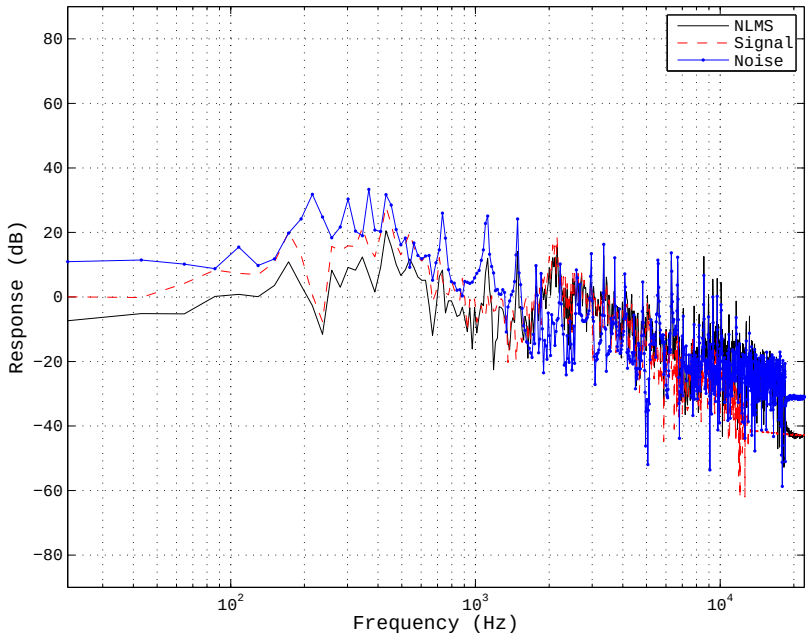


Figure 3.13: Speech in music, without the modification to LMS. -12dB signal to noise ratio.

3.4.5 Summary

The modified NLMS method where additional additive noise is inserted into the system to automatically switch off the adaptive filter in high

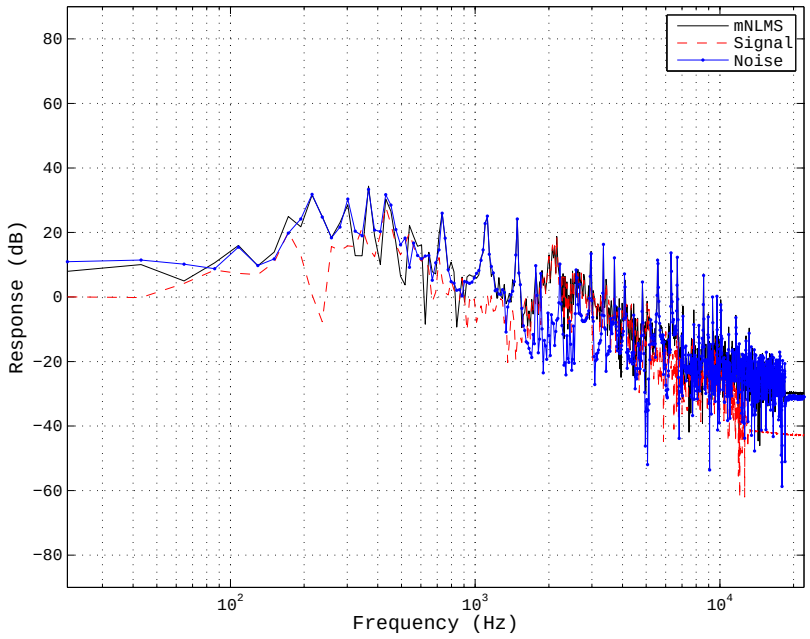


Figure 3.14: Speech in music, with the modification to LMS, -12dB signal to noise ratio

SNR conditions shows excellent performance in retaining the original signal compared with the regular NLMS algorithm in high noise environments. At low SNR, the performance of the modified technique is mixed, showing in general, slightly worse performance than the original NLMS algorithm. The trade-off for this slightly reduced performance at low SNR is its 'hands-off' nature to adaptive filter shut-off — requiring no user intervention to prevent the adaptive filter failing in low noise environments. Further research into improving this technique is required to maintain similar noise cancelling ability as NLMS at low signal to noise ratios.

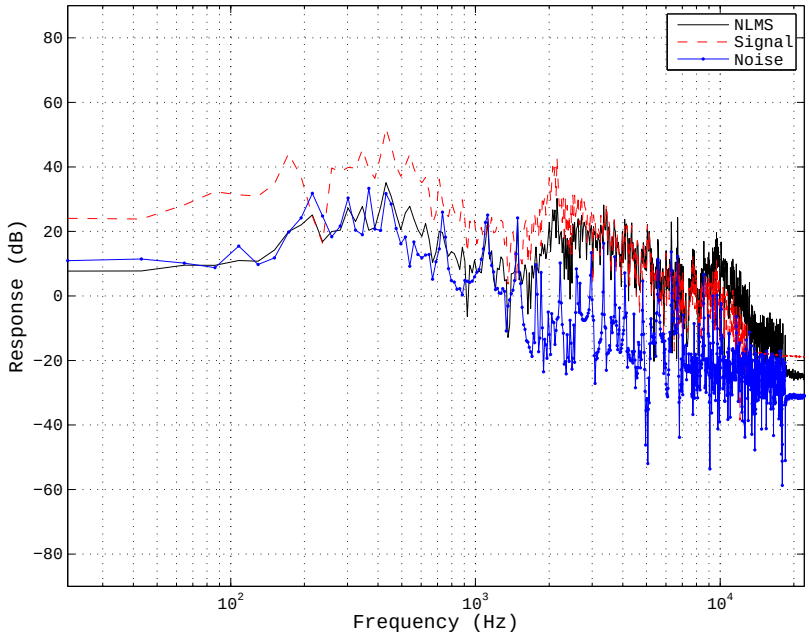


Figure 3.15: Speech in music, without the modification to LMS. 12dB signal to noise ratio.

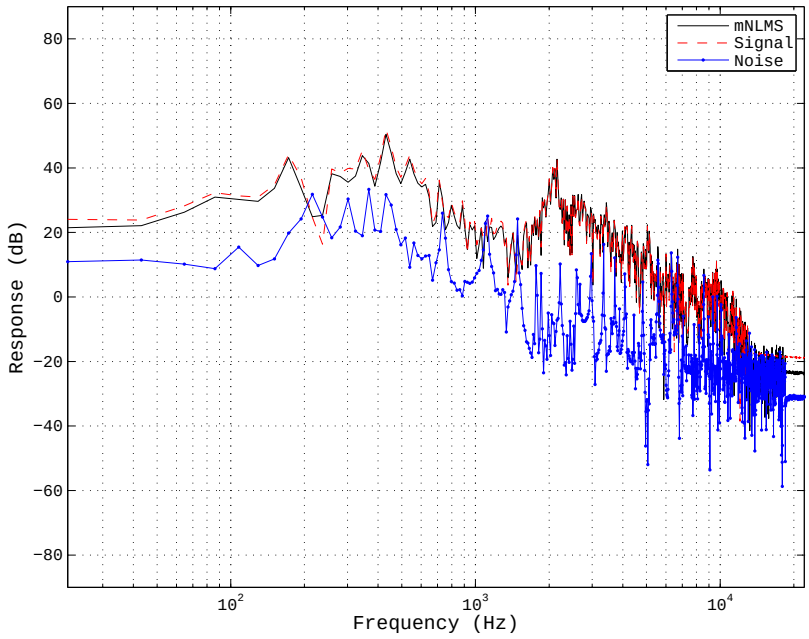


Figure 3.16: Speech in music, with the modification to LMS, 12dB signal to noise ratio. The modified NLMS method retains the original signal.

3.5 Crosstalk Cancellation

The LMS algorithm presented in Section 3.3 exhibits problems in a crosstalk environment where the talker signal leaks into the reference channel, requiring modification to prevent problems arising in high signal to noise conditions (Section 3.4). From Figure 3.4, the filtering system can only use the information from the reference channel to perform any kind of filtering. As an instantaneous, single frequency operation, the LMS system can be expressed in matrix terms (3.1), with the resulting estimate for the noise reduced talker channel presented as the first element in (3.2). If talker-reference crosstalk exists and is significant, which in a near-field situation is generally true, then it is apparent that any simple LMS algorithm will degrade the talker signal (by a factor of $\tilde{H}H_{21}$). The problem arises from the LMS systems inability to simultaneously correct two crosstalk paths. The original adaptive filter noise cancellation system presented in Figure 3.4 essentially cancels the noise-primary crosstalk path, from this it seems reasonable to assume that a second adaptive filter would be required to remove the talker-reference crosstalk in the system. Such systems have been devised in the past [29], [30], [31], [32], [33].

In this section, an implementation of SAD [31] is tested and compared with the simpler LMS algorithms presented in the earlier sections. The algorithm tested was the feedback variant mentioned in the paper. A brief outline follows.

The differences between the instantaneous primary/reference signals

and the filtered previous outputs are computed,

$$e_1 = x_1 - \boldsymbol{y}_2^T \boldsymbol{w}_1$$
$$e_2 = x_2 - \boldsymbol{y}_1^T \boldsymbol{w}_2$$

This stage differs from (N)LMS. In LMS, the error values are computed using the unprocessed inputs from the primary and reference microphones.

The filter updates are given as,

$$\boldsymbol{w}_1 = \boldsymbol{w}_1 + \mu_1 \frac{e_1 \boldsymbol{y}_2}{\boldsymbol{y}_2^T \boldsymbol{y}_2}$$
$$\boldsymbol{w}_2 = \boldsymbol{w}_2 + \mu_2 \frac{e_2 \boldsymbol{y}_1}{\boldsymbol{y}_1^T \boldsymbol{y}_1}$$

Finally the filter inputs are updated,

$$\boldsymbol{y}_1 = [e_1 y_{1,1} y_{1,2} \ldots y_{1,L-1}]^T$$
$$\boldsymbol{y}_2 = [e_2 y_{2,1} y_{2,2} \ldots y_{2,L-1}]^T$$

The key difference between SAD and (N)LMS is the use of processed outputs ($\boldsymbol{y}_1$ and $\boldsymbol{y}_2$) as the input of the next filter step. LMS uses the raw inputs provided from the primary and reference microphones by comparison. This results in faster convergence and effective source separation provided the filters are stable.

3.5.1 Stability

As seen in the simulations, SAD can produce strange artefacts resulting from convergence problems. In practice it was found that good performance (no artefacts) could only be obtained using carefully selected

filter convergence parameters, which unlike NLMS, was found to be difficult to control.

3.5.2 Example: Two Sinusoids

The first simulation involved two sinusoids (49.5Hz and 176.4Hz) in noise at -12dB and 12dB SNR. At -12dB, SAD performs similarly to the modified NLMS technique presented in the previous section. The two signals are clearly visible in the spectrum, with noise still present after filter convergence (Figure 3.17). At 12dB, the instability of the design is visible. Both signals are resolved almost perfectly, however large amounts of musical noise resulting from filter instability is also present (Figure 3.18).

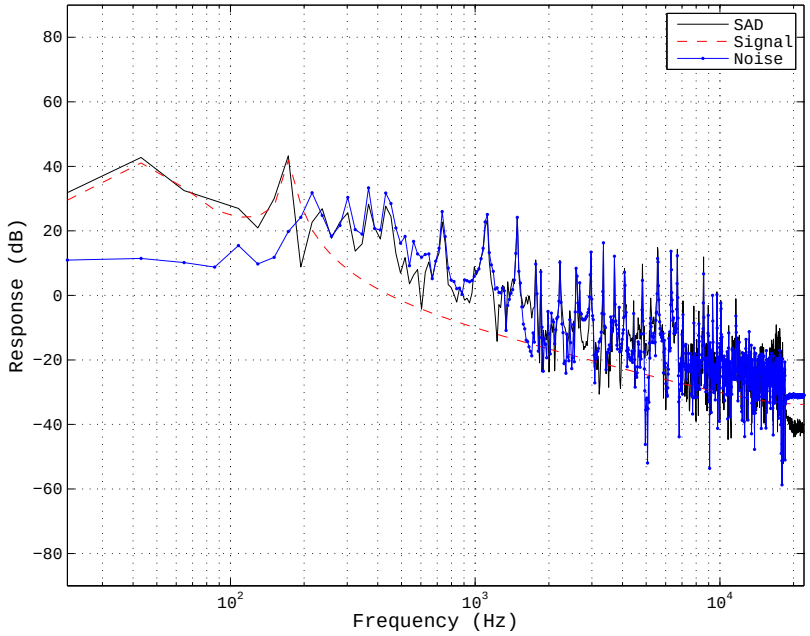


Figure 3.17: Two sine waves in music. -12dB signal to noise ratio. SAD retains the original signal with much of the noise.

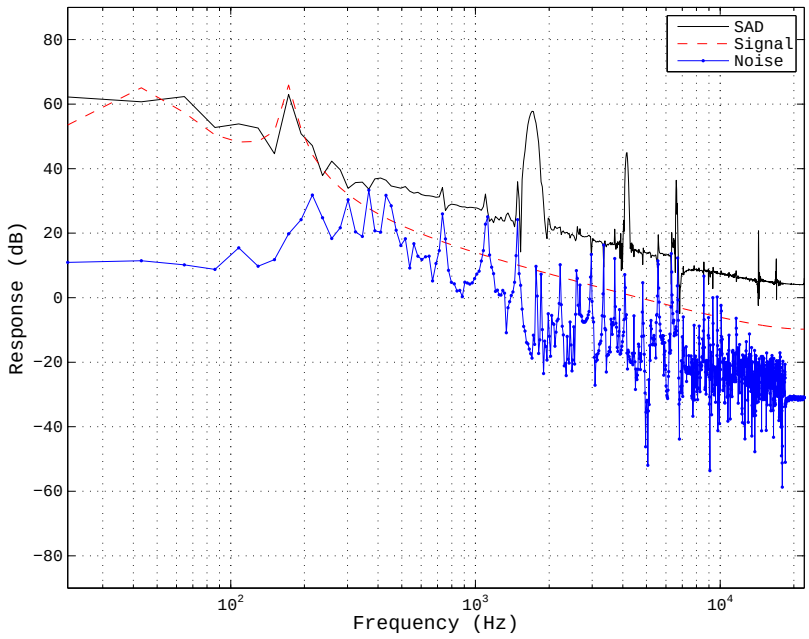


Figure 3.18: Two sine waves in music. 12dB signal to noise ratio. The artifacts arising from poorly controlled convergence are apparent. Several spurious peaks are visible at approximately 1.5kHz, 4kHz and 5.5kHz.

3.5.3 Example: Music

The second simulation involved a music track in noise at -12dB and 12dB SNR (as for the NLMS and modified NLMS cases). Performance of SAD was mixed. Noise reduction was achieved at -12dB SNR (Figure 3.19). However, at 12dB SNR, the algorithm produced a poor output, failing to match the original signal (Figure 3.20).

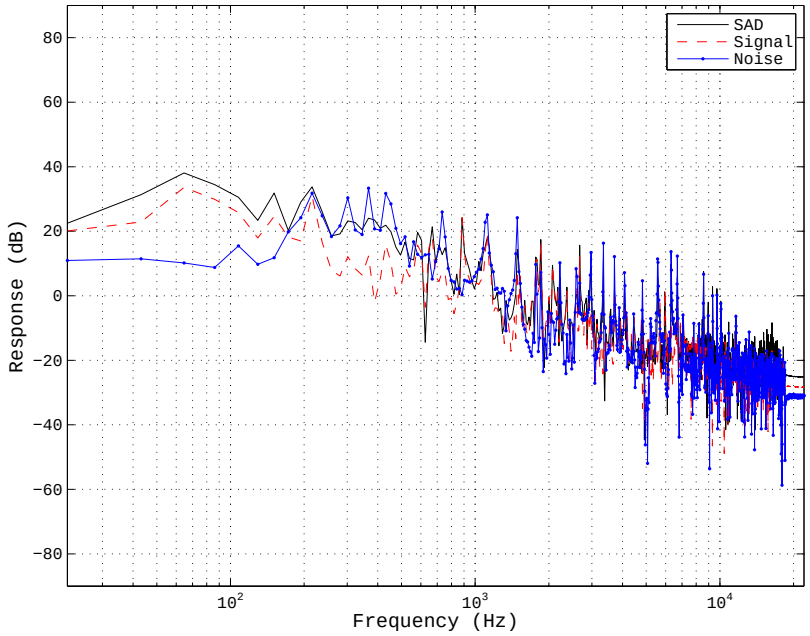


Figure 3.19: A single music track in another music track (noise). -12dB signal to noise ratio. Some noise reduction is occuring, with the spectrum of the output closely following the original spectrum, with some noise present.

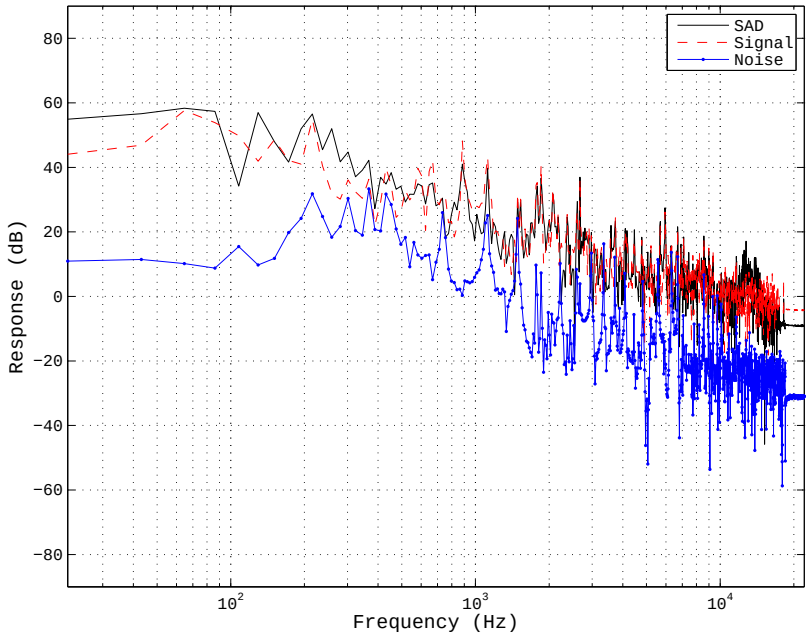


Figure 3.20: A single music track in another music track (noise). 12dB signal to noise ratio. The output fails to match the original signal at lower frequencies. Additionally, a burst of noise is visible at high frequencies (15kHz).

3.5.4 Example: Speech

The final simulation involved speech in noise at -12dB and 12dB SNR. SAD failed to match the original signal at both -12dB SNR. At 12dB SNR, the output very closely matched the original signal, but as was apparent in the other simulations, bursts of high frequency noise occurred.

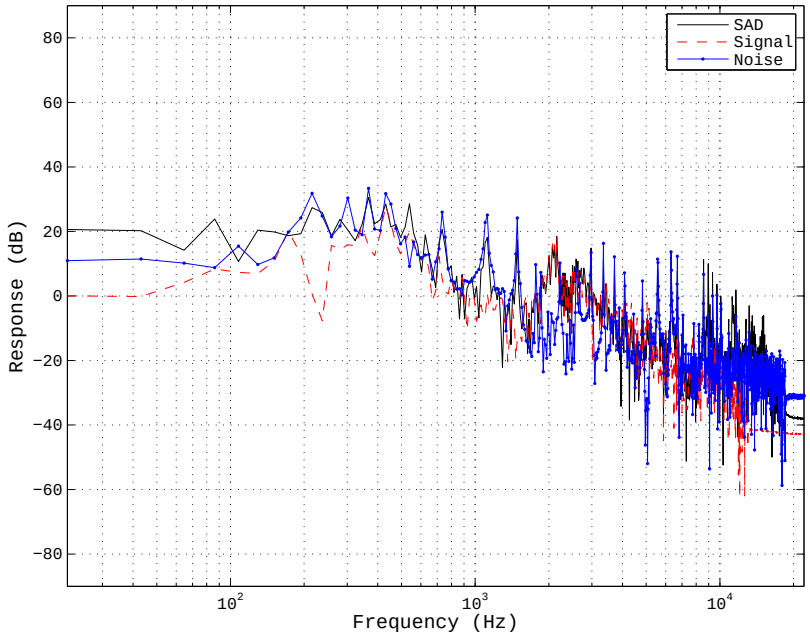


Figure 3.21: Speech in music. -12dB signal to noise ratio. SAD fails to match the original signal at low frequencies.

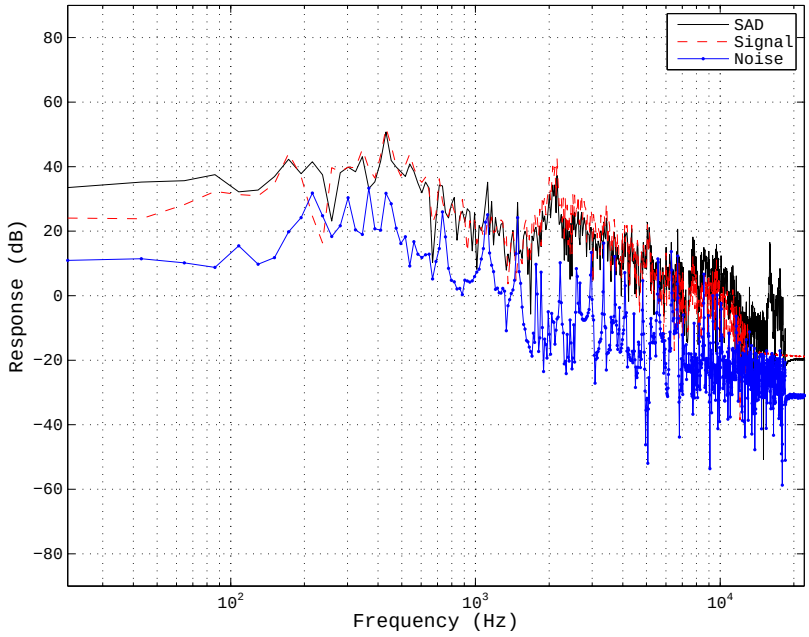


Figure 3.22: Speech in music. 12dB signal to noise ratio. SAD produces a generally good output, except for a small burst of noise at high frequencies.

3.6 Recursive Least Squares Filters

Faster convergence of the least squares problem presented in (3.7) can be obtained through the use of recursive least squares (RLS) filters [11]. The (N)LMS filters presented in Section 3.3 attempt to instantaneously estimate the filter required to reduce the cost function to zero. RLS improves on this technique by extending the computation of the filter to include information from previous iterations. RLS is a more computationally expensive operation, trading scalar-vector multiplications in (N)LMS for vector-matrix multiplications in exchange for faster convergence. In this thesis, this method of adaptive filtering is ignored due to the low complexity requirements imposed.

3.7 Summary

The noise cancelling application of adaptive filtering shows some promise for the purposes of this thesis. In this chapter a simple two microphone noise cancelling example of adaptive filtering (using different filtering algorithms) was presented. The basic NLMS algorithm was evaluated and compared with a modified cross-talk resistant design and a symmetric canceller. The modified cross-talk resistant design performed as expected, removing some noise at low SNR (like the simple NLMS design), while retaining signals at high SNR (where NLMS failed to do so).

The purpose of introducing adaptive filtering was to include a noise reduction system in a dual beamformer set up. In order to assist the adaptive filtering process, the primary channel in the filtering system could be modified from a single microphone to a beamforming array

directed towards the talker with reduced noise characteristics. The reference channel could also be exchanged for a beamforming array in which the background noise is sampled but not the talker (by placing a null at the talker location). Such a design would allow filtering of the talker signal without encountering the issue of talker-reference cross-talk, potentially removing the requirement of a cross-talk resistant filter. However, this case only holds if the beamformers have knowledge of the position of the talker, which is assumed to be located at a fixed point. In reality, the system should be able to manage variations in the talker position, which may re-introduce cross-talk issues if the talker moves outside the null region for example.

Future work in adaptive filtering would include improvements to the modified NLMS technique presented in Section 3.4, to ensure performance parity with NLMS at low signal to noise ratios. Overlooked in this area of research for this thesis was the concept of adaptive beamforming [34][35][36][37][38][39], in which adaptive filters could be utilised to track the talker (or noise signals) to improve speech quality.

Chapter 4

Beamforming plus Adaptive Filtering

In this chapter a number of beamforming techniques introduced in Chapter 2 will be evaluated in conjunction with a noise reduction post-processor based on the modified LMS adaptive filter technique introduced in Chapter 3. A simulation consisting of multiple interfering sources and a talker is devised to perform the evaluations.

4.1 Introduction

The use of post-processing techniques for microphone arrays such as Wiener filtering have been proposed in the past [4]. Post-processing can compensate for limitations in a fixed beamformer (assuming directions of noise or talkers, and effects of robustness modifications to near-field gain/signal to noise ratio) by adapting to conditions in real-time.

When analysing the adaptive filter scenario, the issue of crosstalk between the desired speech signal and the reference (noise) microphone

was raised. One way of dealing with this issue is to design two beamformers for the primary and reference inputs such that only the primary signal contains the desired speech. Such a method requires perfect or near-perfect knowledge of the talker location which would seem to be unrealistic in most scenarios. In this project however, it has been assumed that the talkers location is known (approximately). Consider a cellphone. In order for the cellphone to function properly (in terms of picking up speech), one would expect the users mouth to be located in a region in front of the microphone. A simple model of this would be to assume the mouth is located at a distance $r \pm \delta$, corresponding to lying in a spherical region near the microphone. In addition, an assumption is made on the interferer(s) location(s), it seems reasonable to assume they are not located at the same location as the talker and would normally be located far enough away from the microphones to lie in the far-field response region of the array (though, this is not a necessary condition).

The assumption made in this post-processor design is that the beamformer will pick up noise which is correlated to noise received by the nullformer. By adding the post-processor, the signal received from the nullformer can be used to filter the noise common to both the beamformer channel and the nullformer channel. A highly directional beamformer (such as the second-order array) will not benefit from this particular post-processing technique as the number of interferers in the beamformer channel will be low (and likely to be uncorrelated with the interferers in the nullformer channel). Other designs should see some benefit with the post-processor.

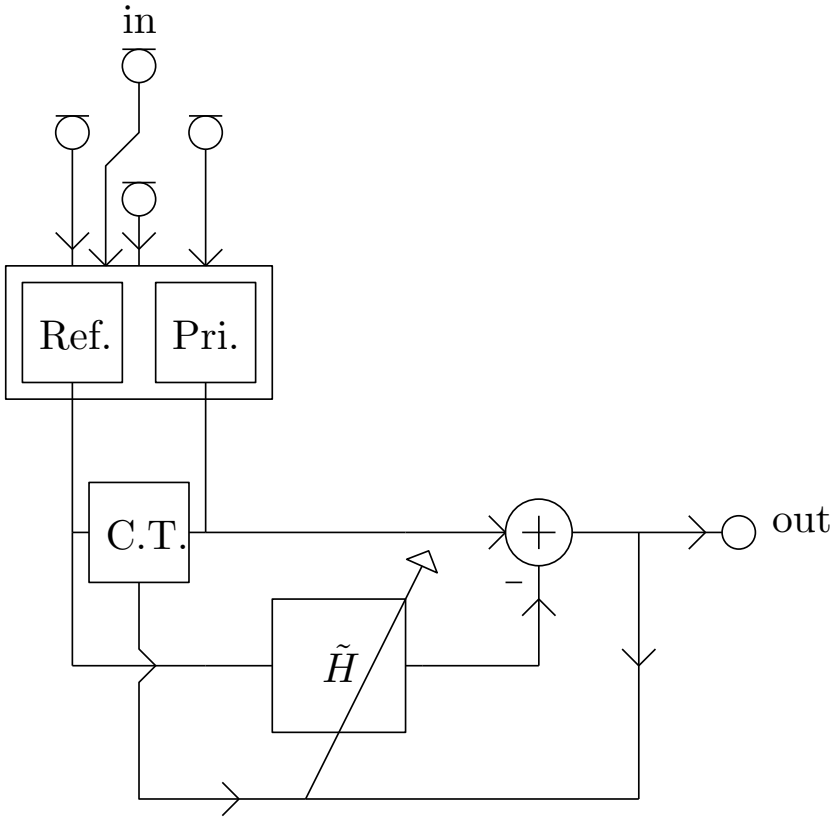


Figure 4.1: System block diagram. Two (or more) microphones fed into dual beamformers and a modified NLMS adaptive filter. The first stage is the beamformer/nullformer FIR filters and equalisation filters (designed for a source located 4cm from the array). The second stage (C.T.) is the cross-talk resistant modification to NLMS (Section 3.4), the two signals are tested for orthogonality. The final stage is the NLMS adaptive filter.

4.1.1 Simulation Setup

The simulation scenario consisted of a source located close to the array (4 cm from the coordinate origin) plus multiple (16) interferers located at random locations. The beam/nullforming weights were calculated for 65 frequencies from zero to the Nyquist rate of the array (44.1kHz to match the sampling rate of the source/interferer files). The inverse DFT was applied to the weight vectors to obtain 128 tap FIR beam/nullforming filters for each of the microphone outputs. Transfer functions (assuming point sources) between the source/interferers were computed and converted into 128 tap FIR filters.

Two simulations are presented for each of the beamformer designs, the first of which consisting of 16 music tracks sampled at 44.1kHz; the second simulation was 10 mechanical interferers (fire truck pump noise, jet fighter cabin noise, factory noise etc.).

The source was a track of speech sampled at 44.1kHz. The signal to noise ratio for the input for the simulations was 0dB.
The input of each microphone was calculated as

$$x_m = \sum_{n=0}^{N} h_n * s_n \tag{4.1}$$

where h_n is the FIR filter describing the transfer function between the source to the microphone and s_n the signal (talker or interferer). The total beam/nullformer output is given as

$$b_1 = \sum_{m=0}^{M-1} w_m^{beam} * x_m \tag{4.2}$$

$$b_2 = \sum_{m=0}^{M-1} w_m^{null} * x_m \tag{4.3}$$

Equalisation filters follow to correct the frequency response for a talker at a specific location. This ensures that the desired talker signal level is identical for each array simulation.

The resulting signals are then fed into an LMS based adaptive filter system, the beamformer (b_1) forming the primary (talker) signal, the nullformer (b_2) forming the reference (noise) signal.

A series of beamformers were tested: no beamforming, differential arrays, matched arrays, maximum near-field gain, variants of least squares methods and a blind source derived method. The nullformer used for the simulations was the ratio least squares method (nulling a 2cm radius region about the talker location). This nullforming technique was chosen for its robustness to positional error and microphone mismatch (good white noise gain), and excellent attenuation of the talker signal (approximately 60dB).

4.2 Omni-directional Reference Array

The simplest array to use in the beamforming and adaptive filtering system is a pair of omnidirectional microphones, one (the closest to the talker) fed into the primary input of the adaptive filter, the other into the reference input. Such a system is not expected to perform well as the signal levels at each of the microphones are nearly identical in high noise environments, provided they are closely spaced. This system, however will provide a baseline for the performance improvements solely due to the adaptive filter component of the system.

The omnidirectional beamforming case is presented in Figure 4.2. The adaptive filter shows some improvement in SNR for the omnidi-

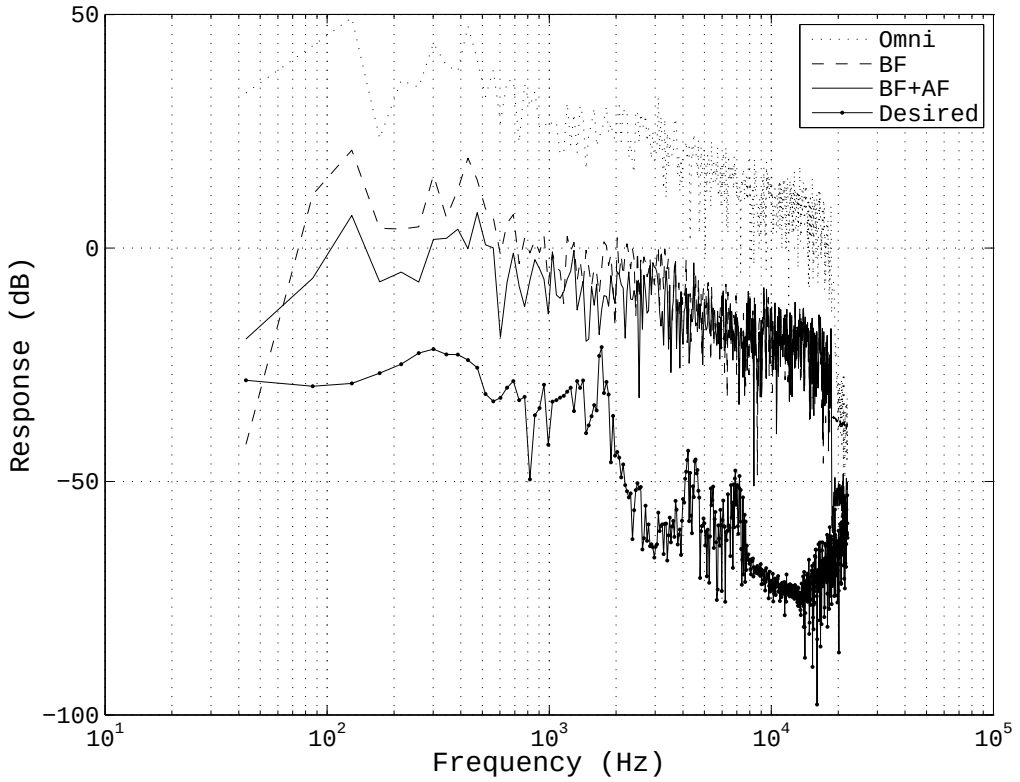


Figure 4.2: Omnidirectional array performance at 0dB SNR in musical noise.

rectional case. The filtered signal shows good reduction in noise level at low frequencies (where most of the speech signal energy lies) with either very small or no improvement at high frequencies.

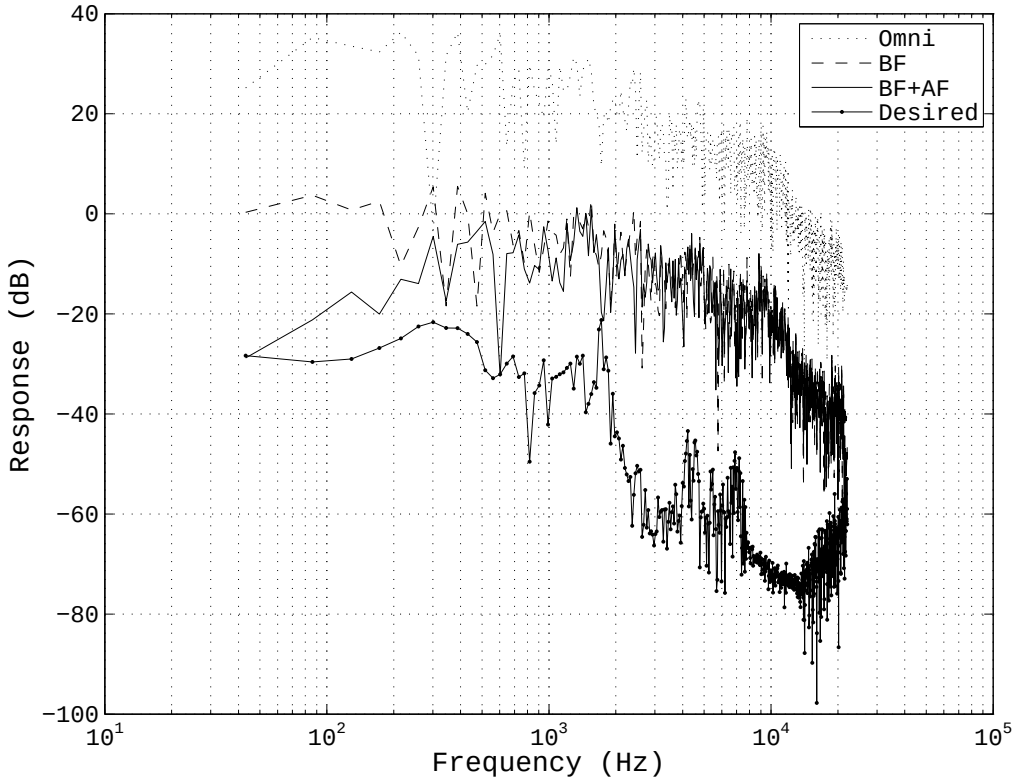


Figure 4.3: Omnidirectional array performance at 0dB SNR in mechanical noise

4.3 Differential Array

First and second order arrays were evaluated in conjunction with adaptive filtering. As derived (and simulated) in chapter 2, higher order arrays exhibit excellent near-field gain corresponding to rejection of far-field interference. The beamformer alone should attenuate far-field sources quite well compared with an omnidirectional microphone without requiring a post processing adaptive filter. However, the talker signal will still contain some interference which could potentially be removed by the adaptive filter.

The first order differential array (Figure 4.4) shows good improvement in signal to noise using only the beamforming technique (the red curve), as expected. Adaptive filtering shows only a moderate improvement in SNR, which, like in the omnidirectional case, occurs mostly at low frequencies.

The second order array (Figure 4.6) shows better performance than the first order case. The beamformed signal shows almost complete rejection of the low frequency components of the interferers. This is expected due to the strong near-field gain effect of the array analysed in Section 2.2.

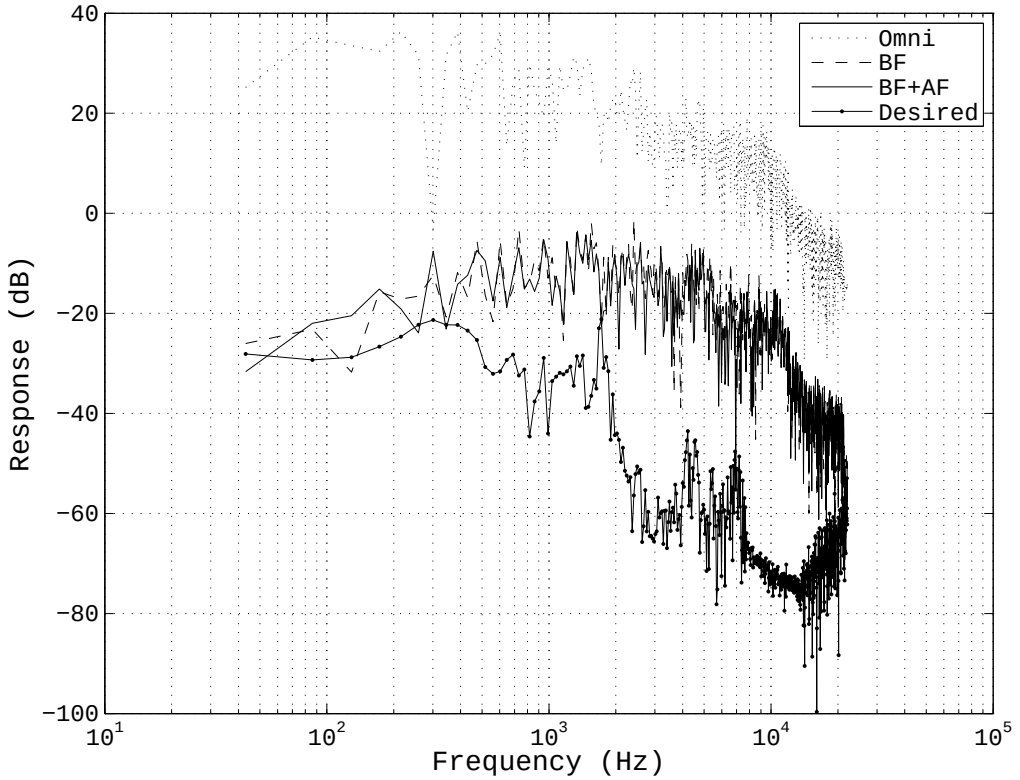


Figure 4.4: First order differential array performance at 0dB SNR in musical noise. The beamforming method delivers most of the improvements in signal to noise ratio, although the adaptive filter removes a significant amount of low frequency noise.

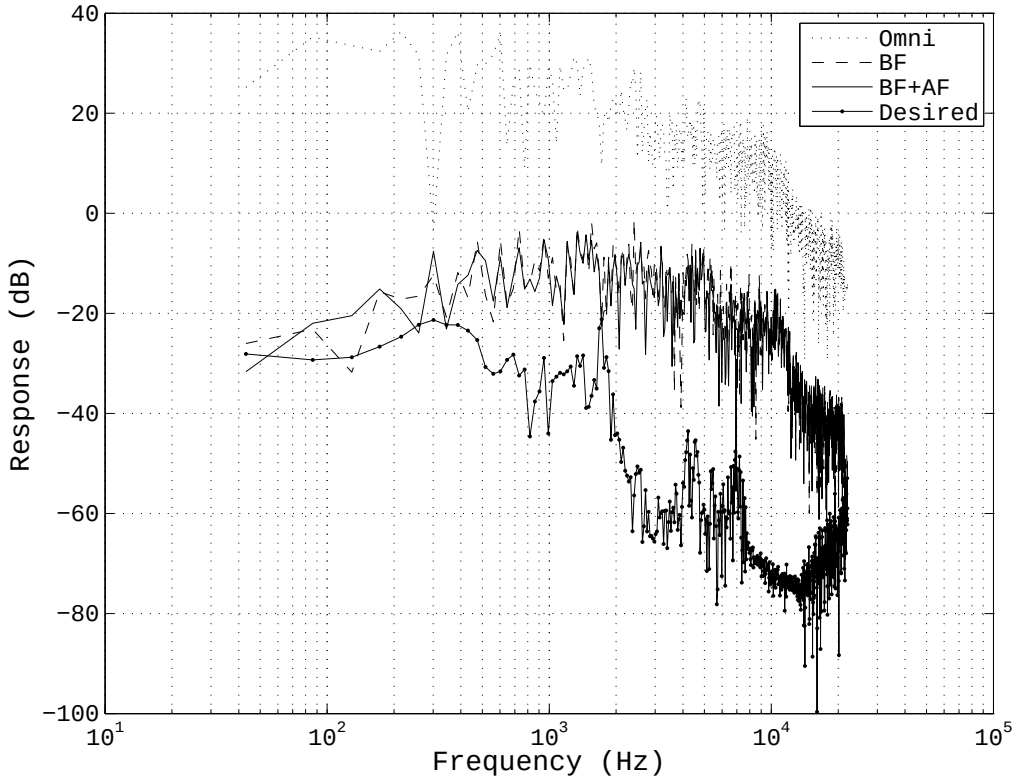


Figure 4.5: First order differential array performance at 0dB SNR in mechanical noise. The adaptive filter reduces noise primarily at mid to high frequencies in contrast to the mostly low frequency improvements in the musical noise simulation.

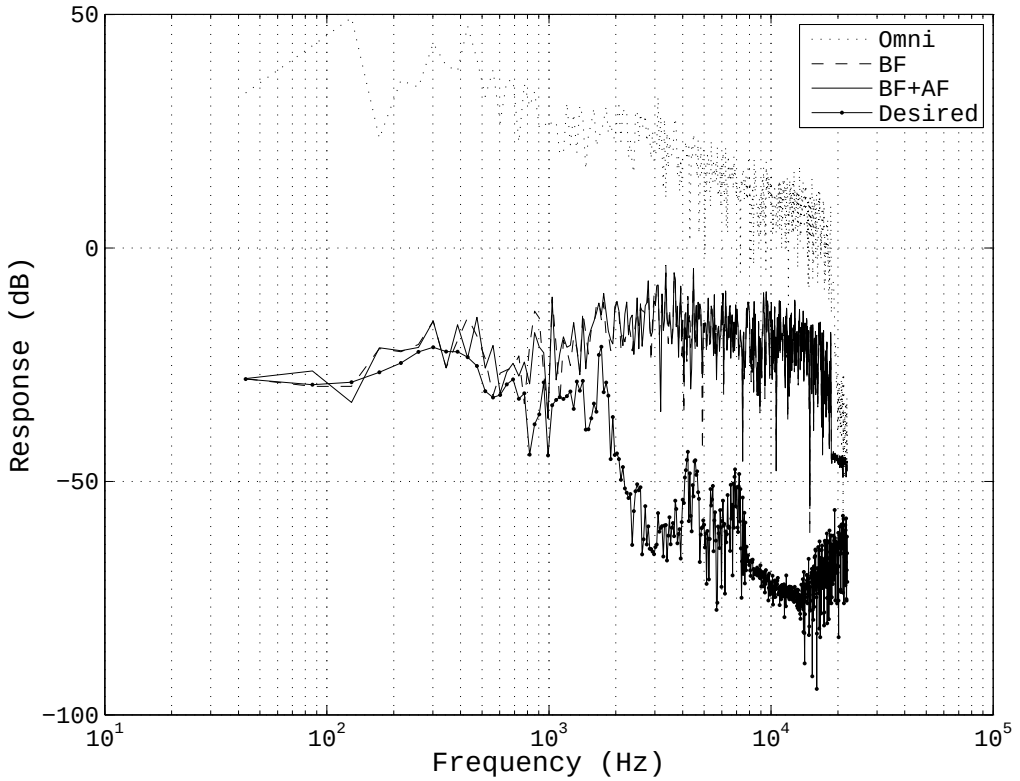


Figure 4.6: Second order differential array performance at 0dB SNR in musical noise. The adaptive filter does little work in reducing noise, the beamformer only and post-processed curves largely overlap.

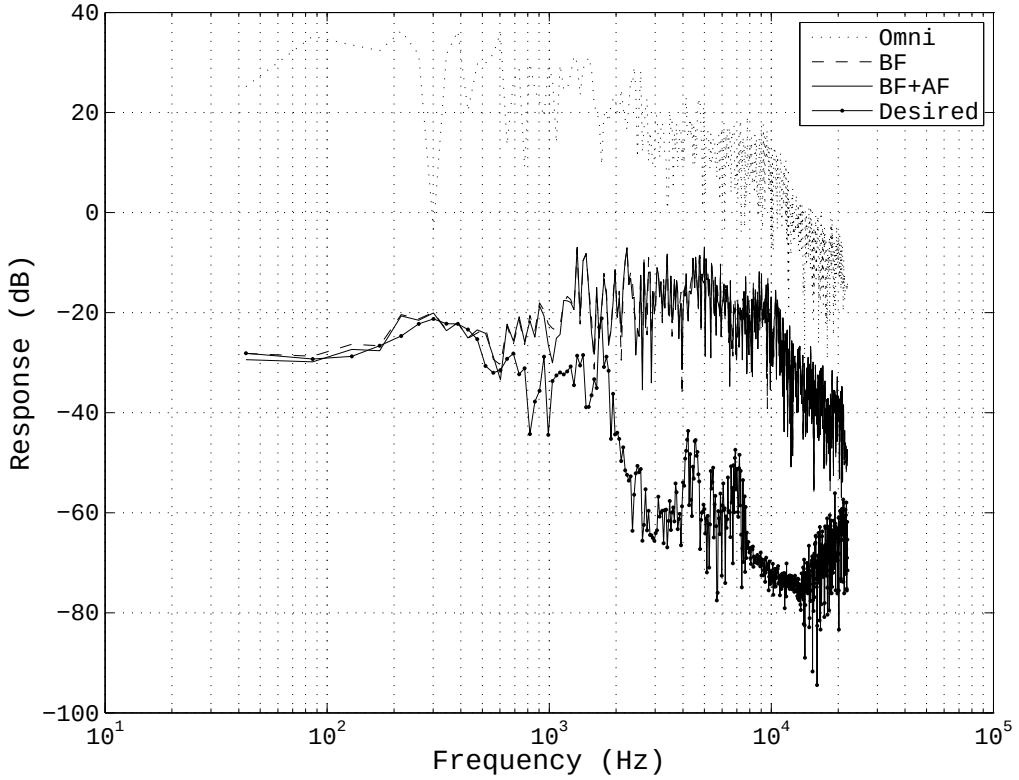


Figure 4.7: Second order differential array performance at 0dB SNR in mechanical noise. The adaptive filter shows little benefit over purely beamforming based methods.

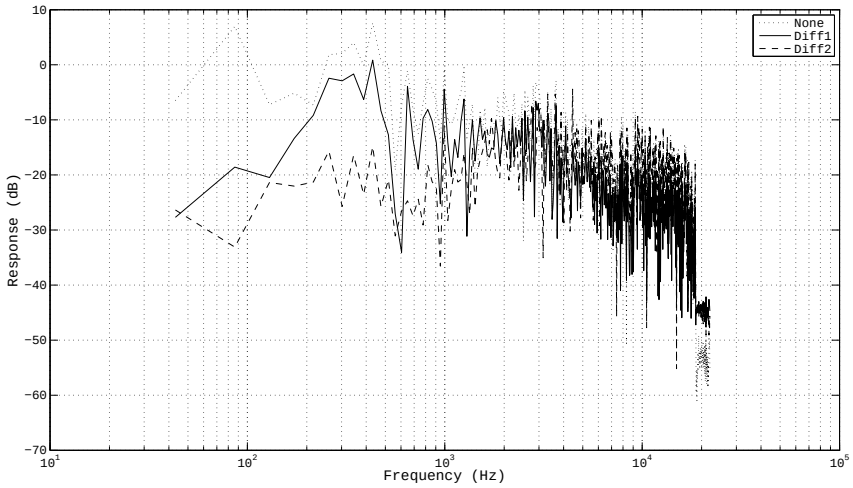


Figure 4.8: Comparison of the 1st and 2nd order designs with the reference (Omnidirectional) microphone in musical noise. The near-field gain advantage of the differential arrays is apparent, particularly at low frequencies.

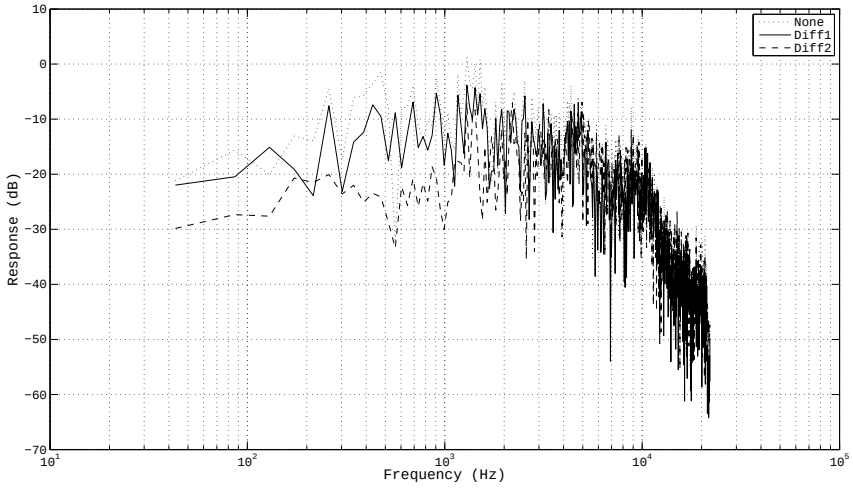


Figure 4.9: Comparison of the 1st and 2nd order designs with the reference (Omnidirectional) microphone in mechanical noise. Both arrays show improved performance over the omnidirectional microphone.

4.4 Near-field Gain Eigenvalue Maximisation (Single Point)

The maximum near-field gain beamformer developed in Section 2.3 was tested. Despite theoretically having the best near-field gain, the actual implementation was limited through the use of a regularisation parameter $\lambda = 10^{-3}$ to ensure robustness (in terms of white noise gain).

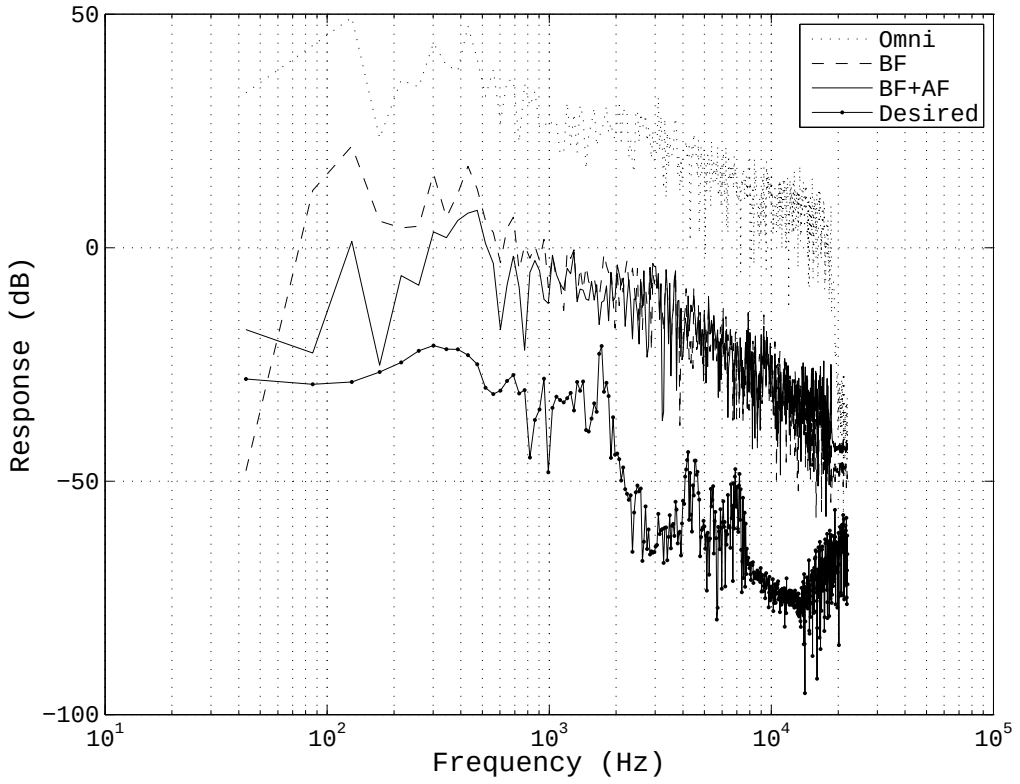


Figure 4.10: Maximum Near-field gain array performance at 0dB SNR in noise. The adaptive filter produces a good reduction in noise from the unprocessed beamformer input.

The maximum near-field gain beamformer was designed to reject far-field sources lying significantly further away from the microphone array than the talker. In these simulations, a number of the interferer

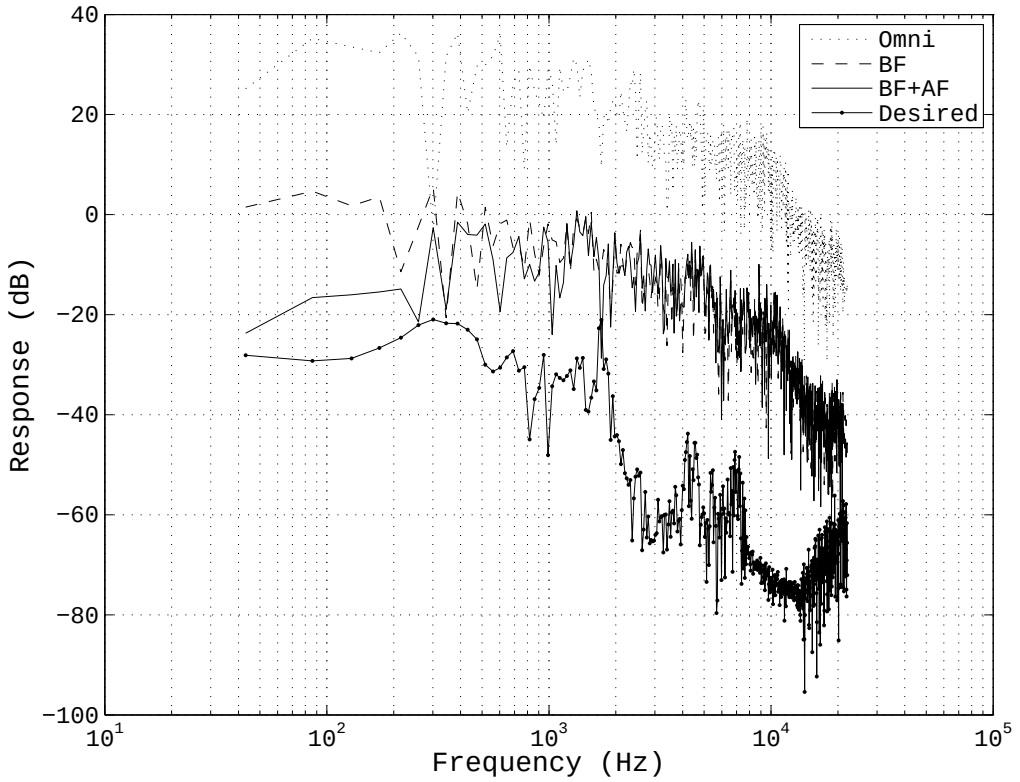


Figure 4.11: Maximum Near-field gain array performance at 0dB SNR in mechanical noise. Good noise reduction is achieved with the adaptive filter post-processor.

signals were set up to be located close to the array (further than the talker but less than 1m from the array). Combined with the limited near-field gain due to regularisation, the near-field gain optimised design performs poorly compared with the other designs.

4.5 Least Squares Beam/Nullforming

The approach in this sub-section is to create two beamformers using least squares solutions for the microphone weights to perform the microphone equivalent of loudspeaker pressure matching. One designed to receive signals from the talker (plus any interferers located in the same direction), the other to place a null in the region around the expected talker location.

The previous methods tested assumed the talker location was fixed exactly, not taking into account movement of the talker or microphone array. The weight solutions (in the differential array and matched filter cases) were not necessarily robust to errors in microphone positions or response. The earlier solutions relied on precise distances between the microphones in the array to the sources. In near-field beamforming, small perturbations in position could have large effects on the array output.

One of the objectives of least squares beamforming is to design weights which are more robust to talker-array distance perturbation as well as tolerating microphone response mismatch (to better reflect reality). The three techniques described in Chapter 2, Section 2.5, are used in these simulations.

4.5.1 Simple LSQ Beamforming

The first technique is as follows: define a pressure vector $\boldsymbol{p}(\boldsymbol{x})$ describing the desired response over a set of points lying on a circle of radius r_0 (the distance between the talker and centre of the array). The pressure vector for the beamformer consists of two separate regions: the region

between -45 deg. and 45 deg. is matched to a series of point sources lying on this segment of the circle; the remainder of the pressure vector is set to zero (no point sources along the rest of the circle).

$$\boldsymbol{p}_b(X(r,\theta)) = \begin{cases} \frac{e^{-ikr}}{r} & \text{if } \theta \in [\frac{-\pi}{4}, \frac{\pi}{4}) \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

$$\boldsymbol{p}_n(X(r,\theta)) = \begin{cases} 0 & \text{if } \theta \in [\frac{-\pi}{4}, \frac{\pi}{4}) \\ \frac{e^{-ikr}}{r} & \text{otherwise} \end{cases} \tag{4.5}$$

The weights are solved using regularised least squares (detailed in Chapter 2).

$$\boldsymbol{w}_{\mathrm{b}} = (H^H H + \lambda I_M)^{-1} H^H \boldsymbol{p}_{\mathrm{b}} \tag{4.6}$$

$$\boldsymbol{w}_{\mathrm{n}} = (H^H H + \lambda I_M)^{-1} H^H \boldsymbol{p}_{\mathrm{n}} \tag{4.7}$$

where λ was set to 10^{-3}.

4.5.2 'Wedge' LSQ Beamforming

The second technique involved extending the matched region to a wedge from $r_0 - \epsilon$ to $r_0 + \epsilon$, the objective of which was to improve robustness of the weight solution, by allowing the talker to move slightly closer or further away from the array without compromising the output significantly.

4.5.3 Simultaneous Maximisation/Minimisation (Ratio) LSQ Beamforming

The third technique was to simultaneously maximise response to the speech signal whilst minimising any signals received from the region

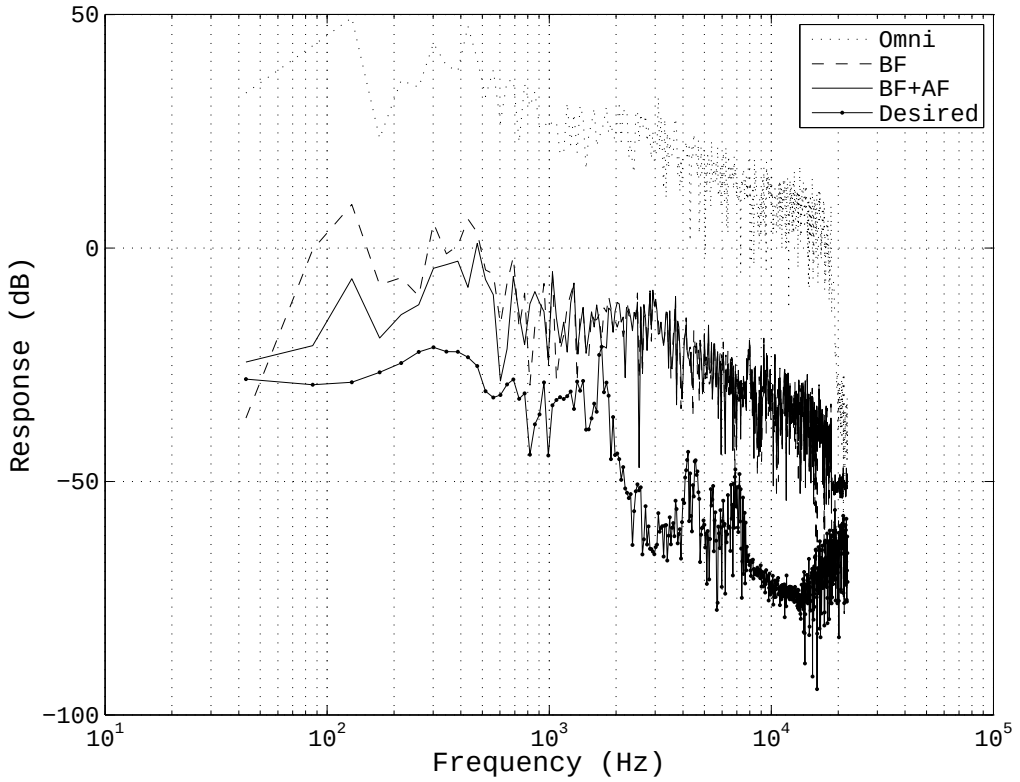


Figure 4.12: Simple LSQ array performance at 0dB SNR in noise. The post-processor reduces noise significantly at low frequencies.

opposite the talker. Two regions are defined: X_1 — a set of positions representing the talker location and X_2 the interferer locations. X_1 was selected as a set of points lying in a 2cm radius circle around the talker location. X_2 was defined as a region of the same area as X_1 placed directly opposite to X_1.

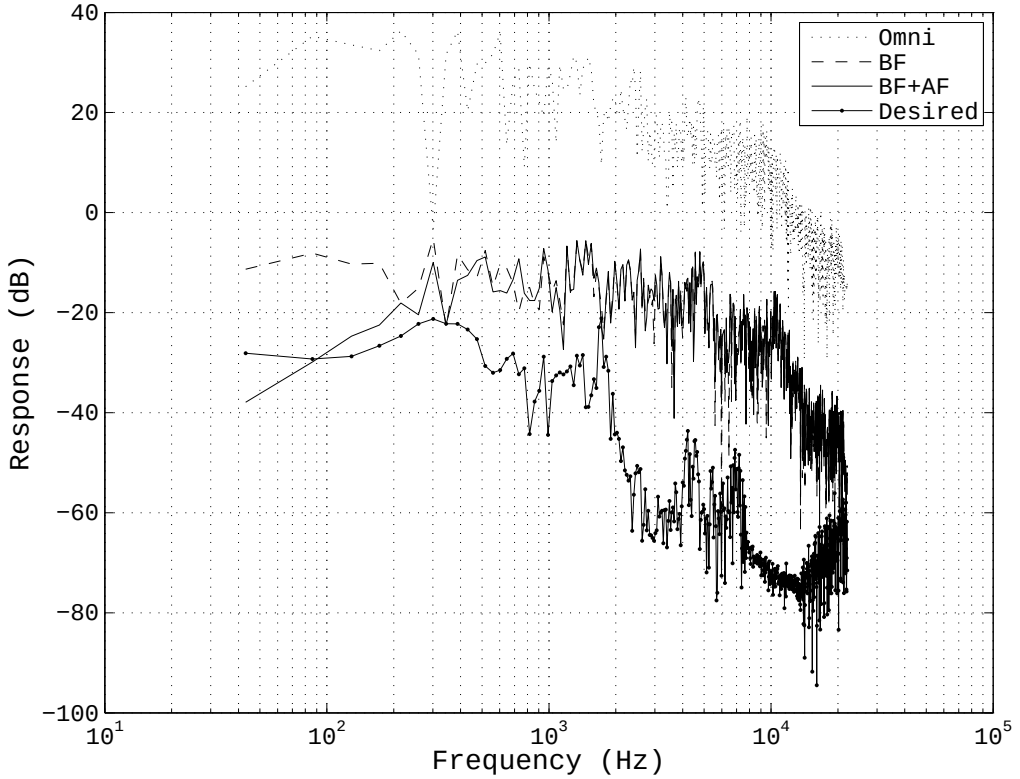


Figure 4.13: Simple LSQ array performance at 0dB SNR in mechanical noise. Noise reduction from the adaptive filter occurs mainly at lower frequencies.

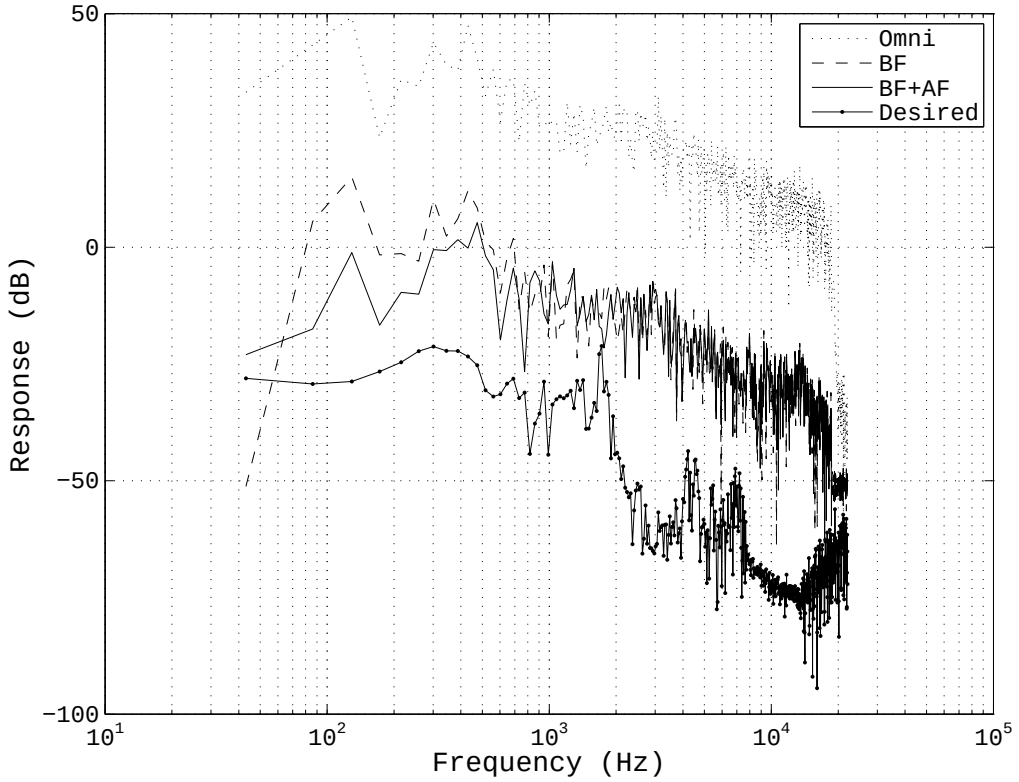


Figure 4.14: Wedge LSQ array performance at 0dB SNR in noise. Overall performance is slightly worse than the simple LSQ based array.

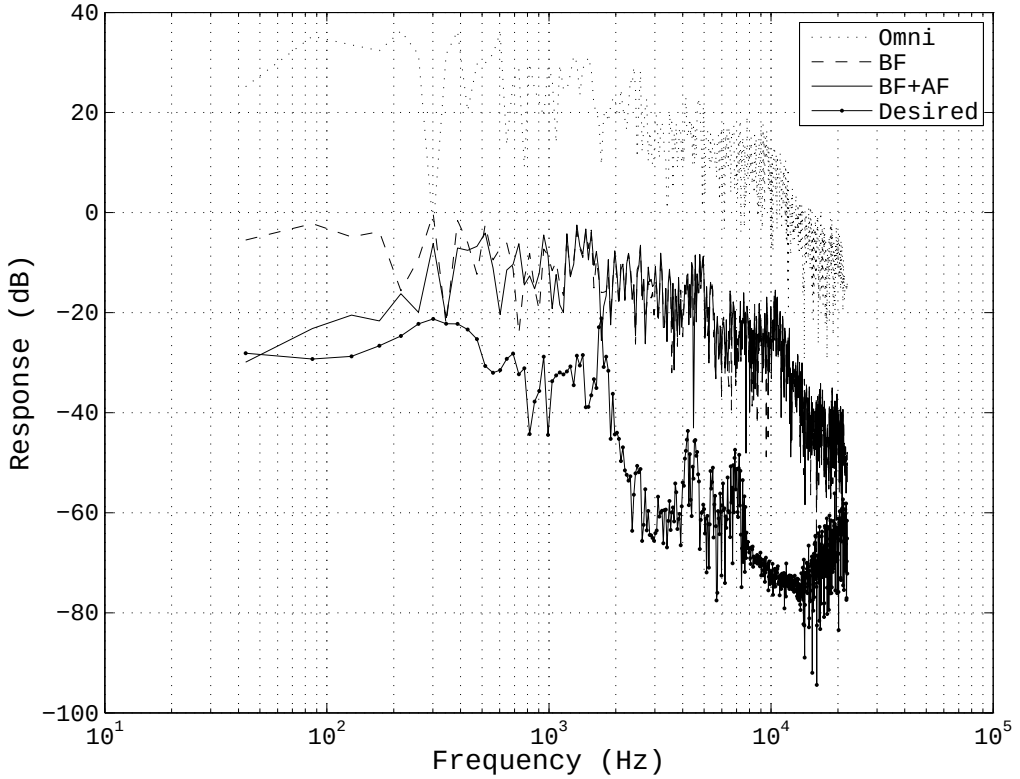


Figure 4.15: Wedge LSQ array performance at 0dB SNR in mechanical noise. The adaptive filter is effective in removing low frequency noise from the inputs.

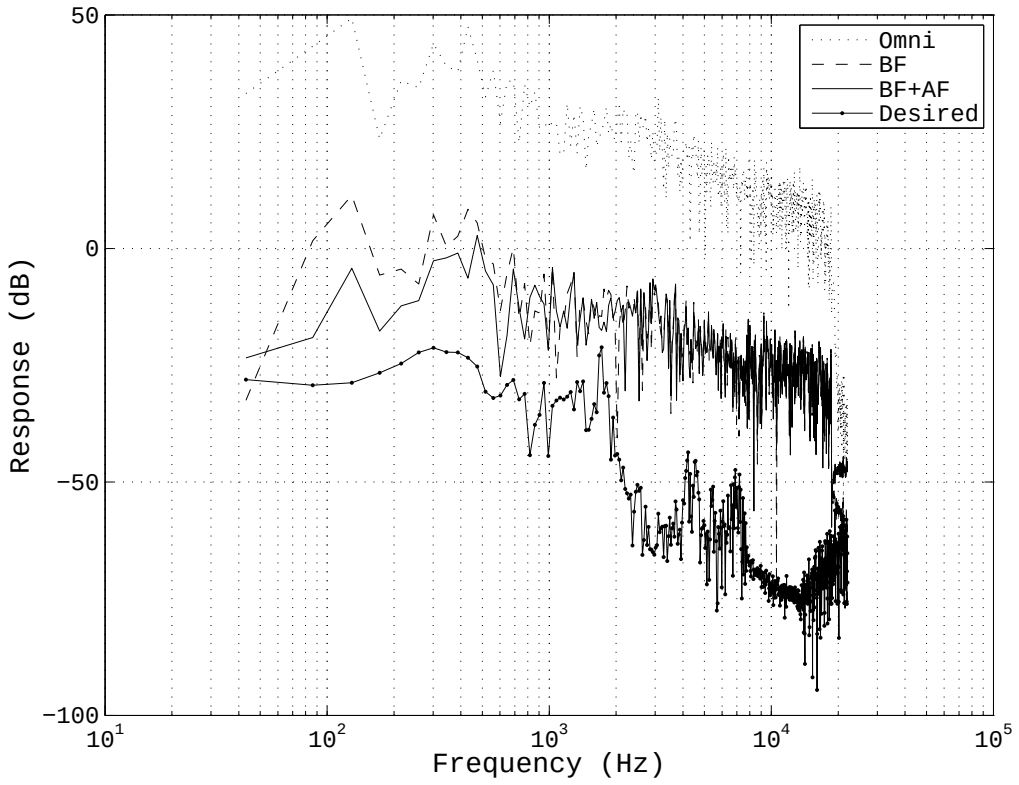


Figure 4.16: Ratio LSQ array performance at 0dB SNR in noise

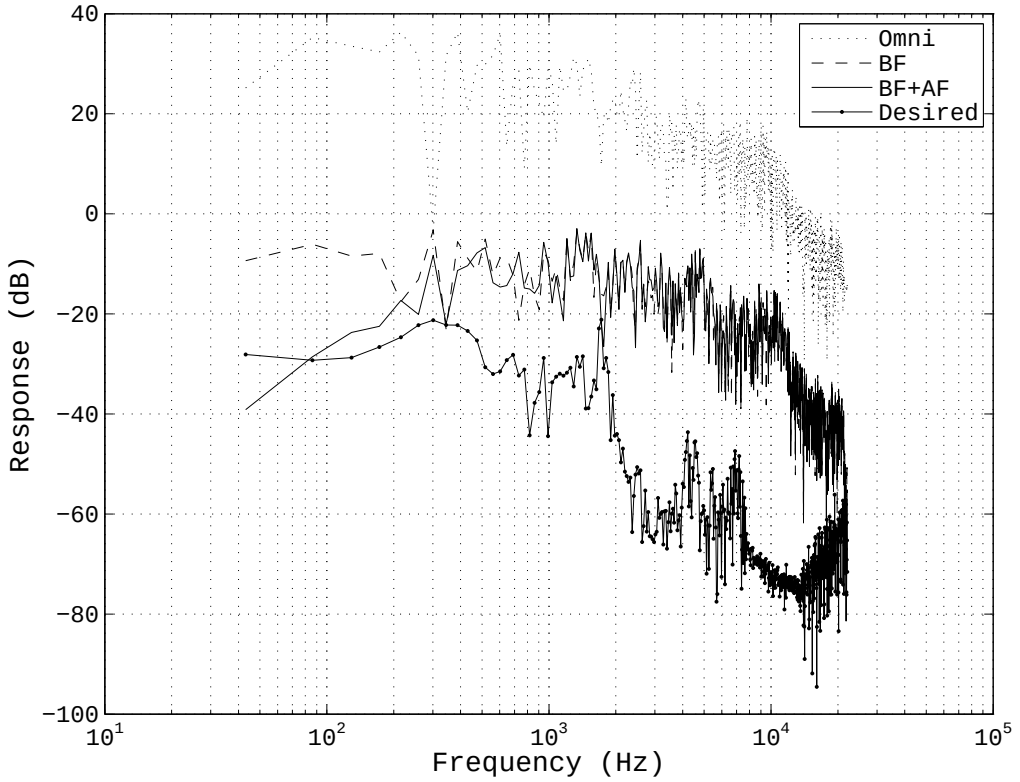


Figure 4.17: Ratio LSQ array performance at 0dB SNR in mechanical noise

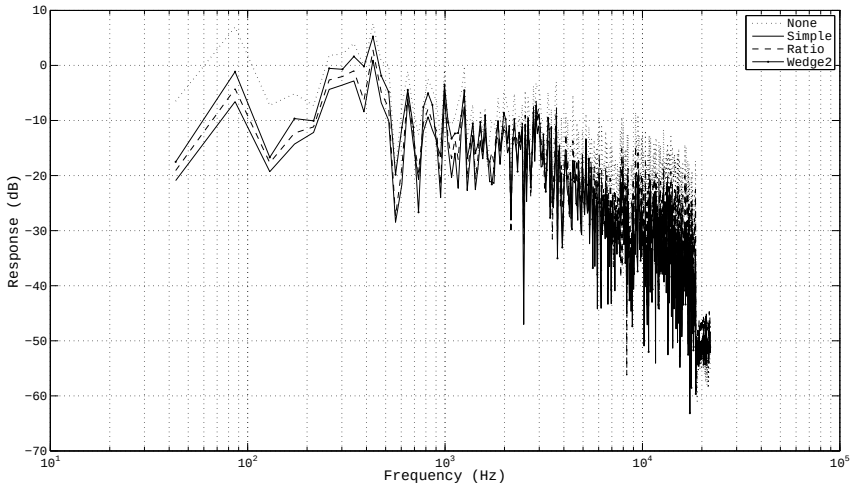


Figure 4.18: Comparison of the three least squares designs and the reference (Omnidirectional) design in musical noise. All array designs show good noise reduction performance when combined with an adaptive filter, particularly at low frequencies.

4.5.4 Comparison of Least Squares Designs

The three least squares techniques are compared with the omnidirectional (purely adaptive filtering based on nulling the talker) approach (Figures 4.18 and 4.19). The trade-offs between talker position robustness and overall performance is apparent. The ratio based technique where the talker is free to move around slightly from the optimal position, shows slightly worse performance in the high frequency regions than the other two designs. In the low frequency region (where the bulk of the speech energy lies) it sits between the two other designs.

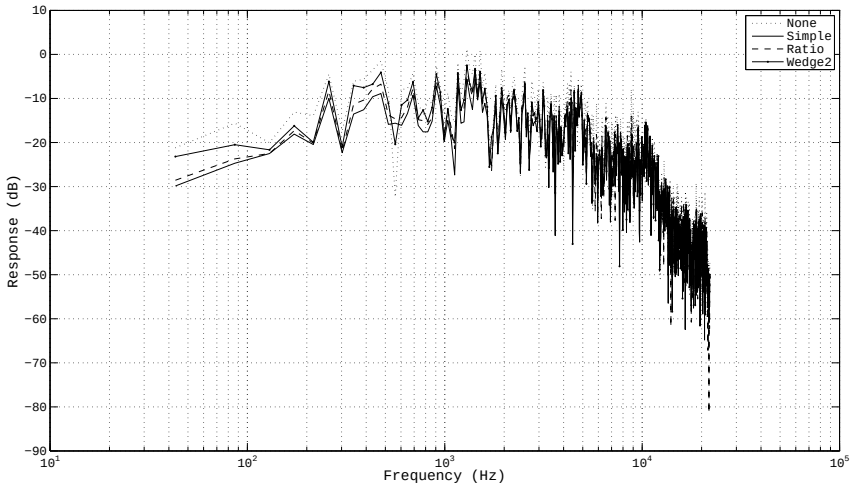


Figure 4.19: Comparison of the three least squares designs and the reference (Omnidirectional) design with mechanical noise. Good low frequency noise reduction is achieved with the post-processor.

4.6 Conclusion

Using the knowledge of various beamforming techniques developed in Chapter 2 and adaptive filtering presented in Chapter 3, a system was devised to attempt to improve the signal to noise ratio of speech (or any near-field signal) in noise. Different beamformers were tested in conjunction with a modified NLMS adaptive filter to find an optimal combination which significantly reduces noise in the output. In Chapter 2, two beamforming techniques were identified which demonstrated potential for enhancing near-field signals — the second order differential array, and a more robust least squares/eigenvalue design. The primary focus of this array design was to investigate whether additional gains in signal to noise ratio could be achieved through the use of a post-processor with these beamforming methods.

All beamforming techniques were evaluated in the post-processing system for academic interest. As mentioned in Chapter 2, the different beamformers have varying levels of near-field gain and white noise gain performance which dictates their suitability for use in an array. The intent of the evaluation was to find case where negative aspects of the beamformers identified earlier were compensated by the inclusion of a post-processor.

The second-order array showed very little improvement when using the post-processor, suggesting that the beamformer is almost ideal from a signal to noise ratio perspective. However, the poor white noise gain highlighted in earlier analysis prevents its use in most situations.

The ratio least squares/eigenvalue design showed some improvement (up to 10dB) at low frequencies when using the post-processor. However, post-processor did not allow the noise reduction performance

to match the second-order array.

Overall, as mentioned in Chapter 2, some sacrifices in beamforming performance must be made in order to produce a practically implementable microphone array. The post-processor was implemented to attempt to compensate for the reduction in near-field gain (and corresponding signal to far-field noise ratio). In all of the beamformer designs (except the second order array), this method was successful at reducing noise, particularly at low frequencies and improved speech intelligibility.

An objective method of measuring speech quality is by evaluating the post-processed signal against the original talker signal through the use of the PESQ algorithm [40]. PESQ is an ITU-T standard detailing a scheme for measuring the voice quality of a signal (after data compression, communication errors or noise reduction) based on a model of perceptual model of human hearing.

PESQ scores obtained on each beamformer/adaptive filter combination (Tables 4.1 and 4.2) show an improvement of at least 0.3 points in intelligibility at 0dB SNR (except the maximum near-field gain eigenvalue design), rendering the speech signal intelligible (although still noisy).

	No Filter	Adaptive Filter
Omni	0.624	0.886
1st Order	1.026	1.060
2nd Order	1.541	1.537
Max NFG	0.674	0.858
LSQ (Simple)	1.092	1.143
LSQ (Wedge)	0.908	1.009
LSQ (Ratio)	0.984	1.048

Table 4.1: PESQ scores for beamformer/adaptive filter combinations with mechanical interferers

	No Filter	Adaptive Filter
Omni	0.643	0.775
1st Order	0.881	1.027
2nd Order	1.497	1.521
Max NFG	0.581	0.746
LSQ (Simple)	0.996	1.153
LSQ (Wedge)	0.744	0.954
LSQ (Ratio)	0.845	1.022

Table 4.2: PESQ scores for beamformer/adaptive filter combinations with musical interferers

Chapter 5

Blind Source Separation

In this chapter blind source separation is introduced briefly for its potential use in extracting speech signals from noisy environments. The FastICA algorithm is introduced and evaluated on a set of noisy signals and its deficiencies detailed. Finally, other blind source separation algorithms that are potentially useful in the future are very briefly described.

5.1 Blind Source Separation

Blind source separation is a set of techniques, including independent component analysis, which separate mixtures of signals generated by some (generally) unknown system. These techniques are useful in many fields of signal processing where information relating to the transmission of signals and the method in which mixtures of signals are produced, are unknown. Applications include telecommunications, medical data (EEG, MRI and others), extracting trends in financial data and image processing [15].

In the speech enhancement scenario presented in Chapter 4, it was

assumed that the system describing the acoustic transfer function of the speech signal were known, by assuming a fixed position of the talker relative to the microphone array. Beamformers were designed to enhance and reject the speech signal for use in an adaptive filtering based post-processor. However, the fixed talker position assumption, key to optimal performance of the beamformer/adaptive filter system presented earlier, is in many scenarios an unrealistic condition. If for example, the microphone array is in a fixed position and the user rotates their head, the beamformer/nullformer is no longer directed at the origin of the speech signal leading to degraded performance of the post-processor.

Blind source separation requires no knowledge of the position of the talker nor the interferers, presenting the potential to improve on the beamforming plus post-processor design by removing the restriction of a fixed talker position.

The signals received at the microphones in the array can be considered to be a result of a beamforming process and an unknown acoustic transfer process. In matrix/vector form, the (instantaneous) microphone signals can be represented by the vector $\boldsymbol{x}$, the result of multiplying the original signals $\boldsymbol{s}$ by the transfer function matrix A followed by the beamforming matrix B, which will be assumed to be an identity matrix, corresponding to omnidirectional responses from the microphones.

$$\boldsymbol{x} = BA\boldsymbol{s} \tag{5.1}$$

The objective of a blind source solution is to find an unmixing matrix, C, such that $C = (BA)^{-1}$. If no initial beamforming is applied to the array ($B = I_M$), then C could be thought of as a beamforming matrix applied to the data in order to separate sources. If beamforming was applied originally, C could be thought of as a transformation matrix applied to

the existing beamforming matrix.

$$C\boldsymbol{x} = CBA\boldsymbol{s} = (BA)^{-1}BA\boldsymbol{s} \tag{5.2}$$

$$= \boldsymbol{s} \tag{5.3}$$

Applying C to the original mixed data set results in a set of estimates of the original unmixed signals.

5.2 FastICA

FastICA [13] is an algorithm developed in the last decade to perform independent component analysis — a particular method of blind source separation, which attempts to separate the individual signals in a mixture by exploiting higher order statistics of the signals. The method separates the signals by finding an unmixing matrix which maximises the non-Gaussianity of the output, through statistical measures such as kurtosis and negentropy and the use of a gradient ascent/descent-like algorithm. The initial required assumption is that the original signals are themselves non-Gaussian.

The first step of the algorithm is to subtract the mean of the individual received signals,

$$\boldsymbol{x} = \boldsymbol{x} - E\{\boldsymbol{x}\} \tag{5.4}$$

The covariance matrix is calculated and its eigenvalue decomposition is taken,

$$C = \boldsymbol{x}\boldsymbol{x}^T = Q\Lambda Q^T \tag{5.5}$$

A rotation matrix corresponding to a whitening process is calculated. This matrix is used to decorrelate the inputs to assist finding solutions

to the unmixing matrix.

$$V = Q^T \Lambda^{-\frac{1}{2}} Q \tag{5.6}$$

Finally, the zero-mean input data is whitened

$$z = V\boldsymbol{x} \tag{5.7}$$

To separate each signal, rows of the unmixing matrix must be estimated. A vector, $\boldsymbol{w}_i$, describing one of the rows of the unmixing matrix is initialised as a random vector and the iterative process begins.

For each iteration, the vector is updated based on an estimate of higher order statistics — kurtosis, negentropy and mutual information minimisation, for example. A contrast function $g(x)$ containing a representation of the higher order statistics and its derivative $g'(x) = \frac{dg(x)}{dx}$ is utilised to iteratively search for the row vector solution $\boldsymbol{w}_i$.

$$\boldsymbol{w}_i = E\{\boldsymbol{z}g(\boldsymbol{w}_i^T\boldsymbol{z})^3\} - E\{g'(\boldsymbol{w}_i^T\boldsymbol{z})\boldsymbol{w}_i\} \tag{5.8}$$

A Gram-Schmidt process of orthonormalisation can be applied to find distinct vectors corresponding to the rows (transposed) of the unmixing matrix to separate up to N signals (where N is the number of microphones).

$$\boldsymbol{w}_i = \boldsymbol{w}_i - \sum_{j=1}^{i-1} (\boldsymbol{w}_i^T \boldsymbol{w}_j)\boldsymbol{w}_j \tag{5.9}$$

$$\boldsymbol{w}_i = \frac{\boldsymbol{w}_i}{\|\boldsymbol{w}_i\|} \tag{5.10}$$

Convergence occurs when the updated vector no longer significantly changes from the result of the previous iteration ($\|\boldsymbol{w} - \boldsymbol{w}_{\text{old}}\| < \delta$). δ defines the minimum change required to continue the iteration process and is typically a very small number, 10^{-3}, for example.

The separated signals can be obtained by computing the product of the unmixing matrix with the whitened signal $\boldsymbol{z}$.

$$\tilde{\boldsymbol{s}} = W^T \boldsymbol{z} = [\boldsymbol{w}_1 \boldsymbol{w}_2 ... \boldsymbol{w}_N]^T V \boldsymbol{x} \tag{5.11}$$

5.2.1 Simulation

Using the base simulation developed for the beamforming and adaptive filtering technique in Chapter 4, FastICA was evaluated on the set of mixed signals — 16 interferers and a talker received by 4 microphones. This is not an ideal simulation for FastICA for two reasons: the total number of signals is greater than the number of microphones, and the mixture is not instantaneous — there are inter-element delays as each signal reaches each microphone.

Kurtosis was used as the measure of non-Gaussianity for finding the unmixing matrix and a deflation scheme using Gram-Schmidt orthogonalisation was used to find four uncorrelated vectors defining the unmixing matrix.

The algorithms are detailed in the original FastICA paper [13] and subsequent material [15][14].

50.7954	50.5413	-50.5976	-50.6792
9.0805	-9.0226	-10.4002	10.3398
-10.2022	10.6846	-10.8408	10.3265
-0.1750	0.0520	1.2737	1.2113

Table 5.1: Converged beamforming matrix. Each row represents a unique beamformer

The objective of the simulation was to find out how effective FastICA operated on a non-ideal set of signals, more signals than microphones, and attempt to find the solution to the unmixing matrix, and associated

beamforming matrix, produced similar results than the beamforming and adaptive filtering based approach presented in Chapter 4.

The resulting beamforming matrix is presented in Table 5.1. Considering the locations of the microphones relative to the source, the first row is almost identical (other than a scaling issue) to the second-order differential array pointed at the source (if the two lateral microphones are considered to be one, as they receive exactly the same signal from the source).

5.2.2 Real-time Processing

The fixed-point FastICA algorithm requires collecting the entire data set before processing, which presents two major problems arising from the requirement that the device store the entire speech segment to perform the processing: first, such a technique requires a large amount of memory to store the noisy signals and secondly, the delay involved in this technique, a large segment of speech must be collected to gather sufficient statistics, which could present a problem if delayed communication is undesired.

The first stage in FastICA and many other ICA algorithms is pre-whitening, a process in which the input signals are decorrelated through an operation involving the computation of the eigenvalues/vectors of the input covariance matrix. In a real-time, on-line scenario, the covariance matrix must be computed at each step and its eigendecomposition found in order to pre-whiten the data. Solving for the eigenvalues and eigenvectors of a matrix is a very computationally expensive operation, seemingly suggesting that attempting to pre-whiten a signal in real-time is difficult.

5.3 Real-time Whitening

Blind source separation algorithms require the data to be prewhitened before the main algorithm acts upon it. Prewhitening involves the computation of the set of eigenvalues and eigenvectors of the correlation matrix in order to compute a change of basis rotation to decorrelate the data. Algorithms for computing the eigenvalues/vectors are in general $O(n^3)$, which suggests that the realtime use of BSS is infeasible at first glance.

The realtime problem involves the computation of a rank-one update to the correlation matrix for each new sample received and the computation of its eigenvalues/vectors, which is a computationally expensive operation. Several techniques have been developed previously for updating matrix decompositions in the past with varying degrees of complexity [41][42][43][44]. In this section, an efficient method for computing the updated eigenvalues and eigenvectors of a real symmetric correlation matrix based on first-order perturbation theory is presented.

A small rank-one correction to the correlation matrix represents a small perturbation to the correlation matrix. Small perturbations to matrices have been studied previously in the context of audio (perturbations to vibrating strings) [45] and notably quantum mechanics [46] (energy levels of an atom in a weak magnetic field, for example). The goal is to find the eigenvalues/vectors of the perturbed system using small corrections based on the eigenvalues/vectors of the unperturbed system.

Starting with a correlation matrix (computed after sufficient lag to ensure it is full-rank), the eigenvalues and eigenvectors can be computed.

$$R = Q\Lambda Q^T \tag{5.12}$$

The update to the correlation matrix involves a rank-one update formed by the outer product of the incoming samples. At time n+1, the update is given as

$$R_{n+1} = \frac{(n-1)}{n}R_n + \frac{1}{n}\boldsymbol{x}\boldsymbol{x}^T \tag{5.13}$$

R_n has an eigendecomposition containing an orthonormal matrix of eigenvectors $Q_n = [\boldsymbol{q}_1\boldsymbol{q}_2...\boldsymbol{q}_N]$ and a diagonal matrix of associated eigenvalues Λ_n satisfying the equation

$$R_n\boldsymbol{q}_{i,n} = \lambda_{i,n}\boldsymbol{q}_{i,n} \tag{5.14}$$

R_{n+1} also has an eigendecomposition $Q_{n+1}\Lambda_{n+1}Q_{n+1}^T$, which will presumed to be a small perturbation to the existing decomposition $Q_n\Lambda_nQ_n^T$. The eigenvalue equation can now be expressed as follows,

$$R_{n+1}\boldsymbol{q}_{i,n+1} = \lambda_{i,n+1}\boldsymbol{q}_{i,n+1} \tag{5.15}$$

$$(R_n + T)(\boldsymbol{q}_{i,n} + \boldsymbol{u}_i) = (\lambda_{i,n} + v_i)(q_{i,n} + \boldsymbol{u}_i) \tag{5.16}$$

where T represents the correction to the autocorrelation matrix, $\boldsymbol{u}_i$ the correction to the i^{th} eigenvector and v_i the correction to the i^{th} eigenvalue. Dropping the n subscripts for clarity,

$$(R + T)(\boldsymbol{q}_i + \boldsymbol{u}_i) = (\lambda_i + v_i)(\boldsymbol{q}_i + \boldsymbol{u}_i) \tag{5.17}$$

Expanding,

$$R\boldsymbol{q}_i + R\boldsymbol{u}_i + T\boldsymbol{q}_i + T\boldsymbol{u}_i = \lambda_i\boldsymbol{q}_i + \lambda_i\boldsymbol{u}_i + v_i\boldsymbol{q}_i + v_i\boldsymbol{u}_i \tag{5.18}$$

Noting that $R\boldsymbol{q}_i = \lambda \boldsymbol{q}_i$ and assuming $T\boldsymbol{u_i}$ and $v_i\boldsymbol{u}_i$ are small enough to ignore,

$$R\boldsymbol{u}_i + T\boldsymbol{q}_i \simeq \lambda_i \boldsymbol{u}_i + v\boldsymbol{q}_i \tag{5.19}$$

Now since the set of $Q = [\boldsymbol{q}_1\boldsymbol{q}_2...\boldsymbol{q}_N]$ forms an orthonormal basis, the update to the eigenvector can be expressed as the weighted sum of each of the existing eigenvectors

$$\boldsymbol{u}_i = \sum_{j=1}^{N} \alpha_j \boldsymbol{q}_j \tag{5.20}$$

$$R\sum_{j=1}^{N} \alpha_j \boldsymbol{q}_j + T\boldsymbol{q}_i = \lambda_i \sum_{j=1}^{N} \alpha_j \boldsymbol{q}_j + v_i \boldsymbol{q}_i \tag{5.21}$$

Putting the R matrix inside the left-hand summation and noting that $R\boldsymbol{q}_j = \lambda_j \boldsymbol{q}_j$,

$$\sum_{j=1}^{N} \alpha_j \lambda_j \boldsymbol{q}_j + T\boldsymbol{q}_i = \lambda_i \sum_{j=1}^{N} \alpha_j \boldsymbol{q}_j + v_i \boldsymbol{q}_i \tag{5.22}$$

Now, by left-multiplying by $\boldsymbol{q}_i^T$ the summations are removed, allowing the computation of the eigenvalue update v_i,

$$\lambda_i \boldsymbol{q}_i^T \alpha_i \boldsymbol{q}_i + \boldsymbol{q}_i^T T\boldsymbol{q}_i = \lambda_i \boldsymbol{q}_i^T \alpha_i \boldsymbol{q}_i + v_i \boldsymbol{q}_i^T \boldsymbol{q}_i \tag{5.23}$$

$$v_i = \frac{\boldsymbol{q}_i^T T \boldsymbol{q}_i}{\boldsymbol{q}_i^T \boldsymbol{q}_i} = \boldsymbol{q}_i^T T \boldsymbol{q}_i \tag{5.24}$$

The eigenvector updates can be found by finding the coefficients α_j of its linear combination of the original orthonormal basis Q. By rearranging (5.22) to place the terms containing $\boldsymbol{u}$ on one side and left multiplying by $\boldsymbol{q}_j^T$ the j^{th} α component can be obtained.

$$\sum_{j=1}^{N} \alpha_j(\lambda_j - \lambda_i)\boldsymbol{q}_j + T\boldsymbol{q}_i = v_i\boldsymbol{q}_i \tag{5.25}$$

$$\alpha_j(\lambda_j - \lambda_i)\boldsymbol{q}_j^T\boldsymbol{q}_j = v\boldsymbol{q}_j^T\boldsymbol{q}_i - \boldsymbol{q}_j^T T\boldsymbol{q}_i \tag{5.26}$$

$$\alpha_j(\lambda_j - \lambda_i) = -\boldsymbol{q}_j^T T\boldsymbol{q}_i \tag{5.27}$$

$$\alpha_j = \begin{cases} \frac{-\boldsymbol{q}_j T\boldsymbol{q}_i}{\lambda_j - \lambda_i} & i \neq j \\ 0 & i = j \end{cases} \tag{5.28}$$

$$\boldsymbol{u}_i = \sum_{j=1}^{N} \alpha_j\boldsymbol{q}_j \tag{5.29}$$

This technique was tested using an existing data set used to develop the beamforming and adaptive filtering techniques. A source located close to a circular microphone array with 16 high power interferers in the far-field. This simulation tests the stability of the estimated eigenvalue and eigenvalue matrices computed using this technique. For each sample the estimated eigenvalues/vectors were computed as well as the full eigenvalue/vector decompositions (using MATLAB's 'eig' function). To compute an initial estimate of the eigenvectors and eigenvalues an initial sample of 1024 samples was taken. As the update technique represents small changes to the matrices, the initial Q and Λ matrices should be computed from a sufficiently large sample size to be representative of the data set.

Figures 5.1 and 5.2 show the matrix norm error in eigenvalue/vector matrices using the perturbation method. Figure 5.1 shows that after an initial fluctuation, the error between the actual eigenvector matrix and the perturbation based estimate remains constant. Analysis of the individual vectors shows that the error results from sign ambiguity in the estimated vectors — two vectors of the four were sign inverted versions

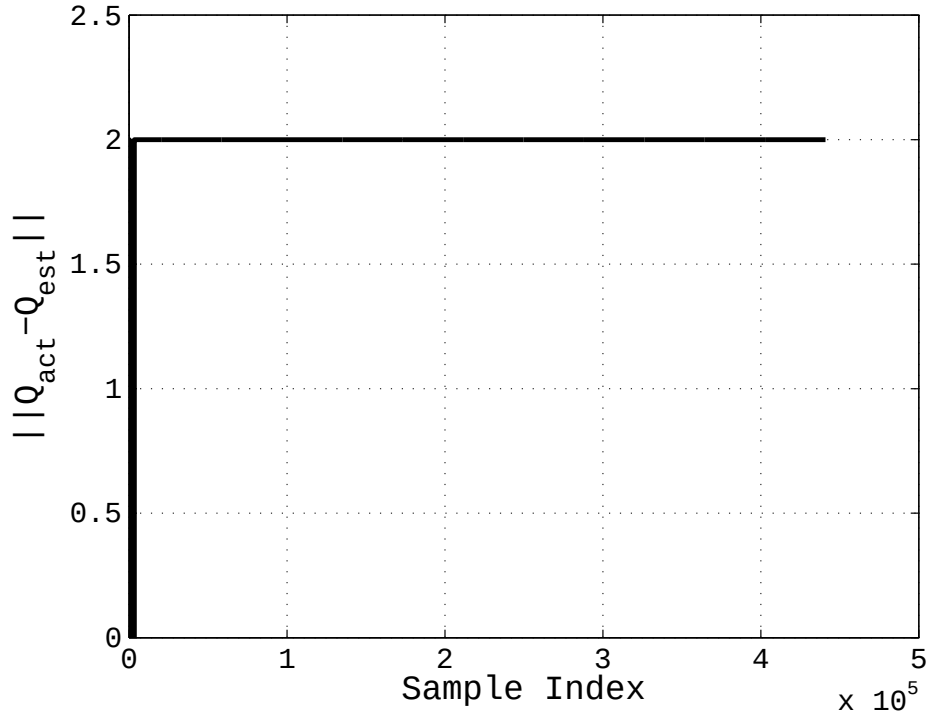


Figure 5.1: Eigenvector tracking error using the perturbation update method. The error, resulting from sign ambiguity, is constant after initial convergence.

of the actual vectors (see Tables 5.2 and 5.3). The error ($\|\Lambda_{act} - \Lambda_{est}\|$) in the eigenvalue matrix is near zero after initial convergence occurring when the original eigenvalues were computed.

-0.5029	0.1980	-0.6773	0.4992
-0.4976	-0.1996	0.6799	0.5003
0.4986	0.6805	0.1969	0.4996
0.5009	-0.6767	-0.2006	0.5009

Table 5.2: Estimated eigenvectors using the perturbation method

The perturbation technique allows good estimates of the eigenvectors/values without requiring a full decomposition for each incoming sample. Eigenvalue decompositions are typically calculated using QR

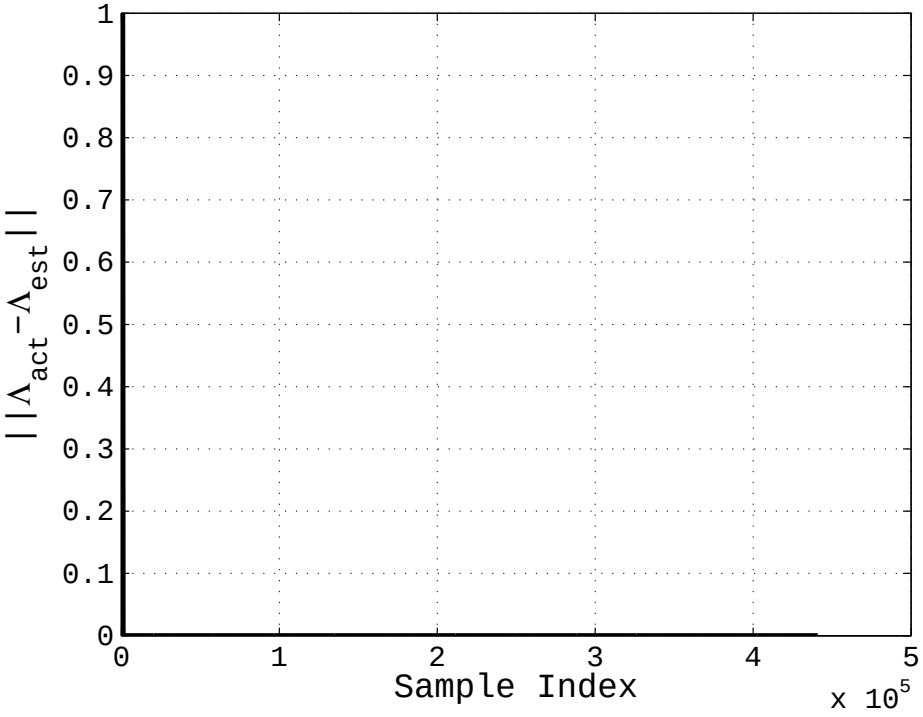


Figure 5.2: Eigenvalue tracking error using the perturbation update method. The error in the eigenvalue matrix is zero after convergence.

0.5029	0.1980	0.6773	0.4992
0.4976	-0.1996	-0.6799	0.5003
-0.4986	0.6805	-0.1969	0.4996
-0.5009	-0.6767	0.2006	0.5009

Table 5.3: Actual eigenvectors

transforms at a computational complexity cost of $\simeq O(n^3)$ [47], preventing their use in low power applications. The perturbation technique requires the computation of an upper triangular matrix containing the correction terms to the eigenvectors and a diagonal correction matrix for the eigenvalues, a computationally simpler method.

5.4 Other Blind Source Separation Algorithms

The FastICA algorithm introduced in Section 5.2 is an example of an instantaneous separation algorithm. The algorithm operated on the assumption that the signals received at each sensor were mixed instantaneously, which was not the case with the simulated data — inter-element delays between sensors were present, preventing the FastICA algorithm running optimally. Convolutive ICA techniques [16][48][49][50] present the opportunity to optimally separate speech signals in a real environment. Problems introduced by inter-element delays between sensors (as was the case in the simulations) and other convolutive effects such as reverberation (not considered in this thesis) could potentially be solved through the use of this class of algorithms.

One of the assumptions under ICA is the number of sensors should match the number of sound sources (desired signal plus interferers). In the simulations earlier this was not the case. The simulations consisted of 17 sources in total with 4 sensors. FastICA could be constrained under this limitation. A set of algorithms for working in an underdetermined (fewer sensors than signals) environment exist [51][52][53] and could potentially be useful in future work in this area.

5.5 Summary

Independent component analysis was briefly presented as a potential technique for speech enhancement. The FastICA algorithm was applied to an existing data set and found to produce a similar level of speech enhancement to the second order differential array plus post-processor technique tested in Chapter 4. A simple technique for reducing the

complexity of the pre-whitening process (part of some ICA algorithms) based on older matrix perturbation techniques was presented a potential first step to reduce the computational complexity of ICA-like algorithms. Finally, a number of more advanced blind source algorithms were identified for future avenues of research.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

The goal of this thesis was to investigate techniques for noise reduction with applications in extreme noise environments. Three areas of research were identified: beamforming, adaptive filtering and blind source separation.

In Chapter 2, two beamformer designs were identified with good characteristics for speech enhancement — the second order differential array, which delivers excellent near-field gain for much of the speech frequency range at the expense of poor robustness to inherent microphone errors; and the ratio least squares design, with good near-field gain (10dB) and good robustness to microphone errors. The least squares design was identified as the optimal design for its balance between near-field gain performance (corresponding to noise reduction) and tolerance to microphone error.

In Chapter 3, basic LMS adaptive filters were investigated in a simple noise cancelling scenario. Cross-talk between the desired signal and

the reference (noise) microphone was identified as a major problem for adaptive filtering in moderate to high signal to noise ratio environments, and a novel technique to compensate was developed.

In Chapter 4, a dual beamforming plus adaptive filter based post-processing system was presented. The beamforming arrays developed in Chapter 2 were tested in conjunction with the modified NLMS adaptive filter system developed in Chapter 3 to evaluate the noise reduction capability of the system. The post-processor was found to improve noise reduction over purely beamforming based methods by a small, though not negligible, level.

Finally, in Chapter 5, blind source separation was briefly introduced as a means of separating speech signals from noise. The FastICA algorithm was investigated on an existing data set prepared for Chapter 4 and found to produce similar results to the second order differential beamformer array. A common pre-processing technique was identified as a potential issue in implementing a FastICA-like algorithm on a low power device and a algorithmically simpler method of performing this pre-processing step was developed. Alternative algorithms more suited to speech enhancement were (very) briefly investigated and will potentially form the basis of future work.

6.2 Future Work

The key assumption in the development of the post-processing system was the (near) perfect knowledge of the talker position — by placing the device as close to the mouth as possible, the distance from the mouth to the microphones is reasonably well known. This assumption allows

the design of a beamforming array to attempt to isolate sound from this position, and a complementary nullformer to reject the talker for use as a noise reference in the post-processor. However, the assumption places a constraint on how the device is used. It may not be possible to use the device in the designed manner at all times, limiting its use to specific scenarios where the talker position is exactly known.

Future work to improve robustness of this design could be to include additional information relating to the environment. The addition of sensors to track movement of device relative to the users mouth could potentially be a method of improving the design of the system, as an example. Sensors could feed information into a beamforming controller, adjusting the weights of the microphones in the array to follow the mouth, allowing the post-processing design to function as normal.

More advanced filtering methods based on multichannel Wiener filtering present future avenues of research. Multichannel Wiener filtering has been identified as a powerful technique for noise reduction in multiple microphone arrays [54][55].

Blind source separation introduced in Chapter 5 potentially solves this problem without requiring additional information from the environment. The advantage of this technique lies in its 'blind' nature. It requires no knowledge of the environment to function — making no assumptions on the position of the talker relative to the microphone array, for example. This thesis briefly investigated basic properties of blind source separation. Future work would extend this to investigate current techniques, in particular for the application to speech enhancement, and investigate efficient implementations of this class of algorithms for low power devices.

Bibliography

[1] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 27, no. 2, pp. 113 – 120, apr 1979.

[2] B. yin Xia, Y. Liang, and C. chun Bao, "A modified spectral subtraction method for speech enhancement based on masking property of human auditory system," in *Wireless Communications Signal Processing, 2009. WCSP 2009. International Conference on*, nov. 2009, pp. 1 –5.

[3] B. yin Xia, C. chun Bao, and Y. Liang, "A fast convergence speech enhancement method," in *Proceedings of the Second APSIPA Annual Summit and Conference*, 2010.

[4] R. Zelinski, "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms," in *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, apr 1988, pp. 2578 –2581 vol.5.

[5] H. F. Olson, "Electroacoustic signal translating apparatus," 1941.

[6] H. E. Ellithorn, "Antinoise characteristics of differential microphones," *Proceedings of the I.R.E.*, 1946.

[7] H. Olson, *Elements of Acoustic Engineering*. Van Nostrand, 1947.

[8] G. W. Elko, "Microphone array systems for hands-free telecommunication," *Speech Communication*, 1996.

[9] S. L. Gay and J. Benesty, Eds., *Acoustic Signal Processing for Telecommunication*. Kluwer Academic Publishers, 2001.

[10] M. M. Sondhi and A. J. Presti, "A self-adaptive echo canceller," *B.S.T.J. Briefs*, 1966.

[11] S. Haykin, *Adaptive Filter Theory*. Prentice Hall, 1991.

[12] B. Widrow and S. Stearns, *Adaptive Signal Processing*. Prentice Hall, 1985.

[13] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis." *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18252563

[14] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, ser. Adaptive and Learning Systems for Signal Processing, Communications, and Control. J. Wiley, 2001. [Online]. Available: http://books.google.co.nz/books?id=96D0ypDwAkkC

[15] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Networks*, 2000.

[16] H. Buchner, R. Aichner, and W. Kellermann, "Trinicon: A versatile framework for multichannel blind signal processing," in *in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2004, pp. 889–892.

[17] J. Ryan and R. Goubran, "Near-field beamforming for microphone arrays," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, vol. 1, apr 1997, pp. 363 –366 vol.1.

[18] I. McCowan, "Robust speech recognition using microphone arrays," Ph.D. dissertation, Queensland University of Technology, Australia, 2001.

[19] D. Colton and R. Kress, *Inverse Acoustic and Electromagnetic Scattering Theory*. Springer, 1998.

[20] G. Arfken and H. Weber, *Mathematical Methods for Physicists*. Academic Press, 1985.

[21] M. Abramowitz and I. Stegun, Eds., *Handbook of Mathematical Functions*. Dover Publications, 1972.

[22] T. Betlehem, C. Anderson, and M. A. Poletti, "A directional loudspeaker array for surround sound in reverberant rooms," *Proceedings of 20th International Congress on Acoustics*, 2010.

[23] Y. Zhao and R. Langley, "A least squares approach to the design of frequency invariant beamformers," *17th European Signal Processing Conference*, 2009.

[24] L. C. Parra, "Least squares frequency-invariant beamforming," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005.

[25] A. Tikhonov, *Numerical Methods for the Solution of Ill-Posed Problems*, ser. Mathematics and Its Applications.

Kluwer Academic Publishers, 1995. [Online]. Available: http://books.google.co.nz/books?id=rpdAzBsOSMgC

[26] A. B. and M. Schroder, "Adaptive predictive coding of speech signals," *Bell System Technical Journal*, 1970.

[27] G. Motta, J. Storer, and B. Carpentieri, "Adaptive linear prediction lossless image coding," in *Data Compression Conference, 1999. Proceedings. DCC '99*, mar 1999, pp. 491 –500.

[28] A. Cauchy, "Méthode générale pour la résolution des systemes d'équations simultanées," *Comp. Rend. Sci. Paris*, vol. 25, no. 1847, pp. 536–538, 1847.

[29] J. Zinser, R., G. Mirchandani, and J. Evans, "Some experimental and theoretical results using a new adaptive filter structure for noise cancellation in the presence of crosstalk," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, vol. 10, apr 1985, pp. 1253 – 1256.

[30] G. Mirchandani, J. Zinser, R.L., and J. Evans, "A new adaptive noise cancellation scheme in the presence of crosstalk [speech signals]," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 39, no. 10, pp. 681 –694, oct 1992.

[31] S. Van Gerven and D. Van Compernolle, "Signal separation by symmetric adaptive decorrelation: stability, convergence, and uniqueness," *Signal Processing, IEEE Transactions on*, vol. 43, no. 7, pp. 1602 –1612, jul 1995.

[32] L. Lepauloux, P. Scalart, and C. Marro, "Low distorsion decoupled crosstalk resistant adaptive noise canceller," in

11th IEEE International Workshop on Acoustic Echo and Noise Control, Seattle, United States, Sep. 2008. [Online]. Available: http://hal.inria.fr/inria-00451048

[33] ——, "An efficient low-complexity algorithm for crosstalk-resistant adaptive noise canceller," in *17th European Signal Processing Conference*. Glasgow, Royaume-Uni: eurasip, Aug. 2009. [Online]. Available: http://hal.inria.fr/inria-00450770

[34] W. Herbordt and W. Kellermann, "Adaptive beamforming for audio signal acquisition," 2003.

[35] N. Jablon, "Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections," *Antennas and Propagation, IEEE Transactions on*, vol. 34, no. 8, pp. 996 – 1012, aug 1986.

[36] S. Hayward, "Adaptive beamforming for rapidly moving arrays," in *Radar, 1996. Proceedings., CIE International Conference of*, oct 1996, pp. 480 –483.

[37] W. Herbordt, "Combination of robust adaptive beamforming with acoustic echo cancellation for acoustic human/machine interfaces," Ph.D. dissertation, Erlangen-Nürnberg, 2003.

[38] L. J. Griffiths, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Transactions on Antennas and Propagation*, 1982.

[39] O. L. Frost, "An algorithm for linearly constrained adaptive array processing," in *Proceedings of the IEEE*, aug 1972.

[40] I. T. UNION, "Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," ITU-T, Tech. Rep., 2001.

[41] P. Strobach, "Fast recursive subspace adaptive esprit algorithms," *IEEE Transactions on Signal Processing*, 1998.

[42] T. Chonavel, B. Champagne, and C. Riou, "Fast adaptive eigenvalue decomposition: A maximum likelyhood approach," *Signal Processing*, 2003.

[43] P. Comon and G. H. Golub, "Tracking a few extreme singular values and vectors in signal processing," in *Proceedings of the IEEE*, 1990.

[44] P. Strobach, "The fast recursive row-householder subspace tracking algorithm," *Signal Processing*, 2009.

[45] J. S. Rayleigh, *The Theory of Sound vol. I.* Macmillan, 1877.

[46] E. Schrödinger, "Quantisierung als eigenwertproblem," *Annalen der Physik*, vol. 385, no. 13, pp. 437–490, 1926. [Online]. Available: http://dx.doi.org/10.1002/andp.19263851302

[47] J. Demmel, *Applied Numerical Linear Algebra*, ser. Miscellaneous Bks. Society for Industrial and Applied Mathematics, 1997. [Online]. Available: http://books.google.co.nz/books?id=lr8cFi-YWnIC

[48] E. Bingham and A. Hyvärinen, "A fast fixed-point algorithm for independent component analysis of complex valued signals," *International Journal of Neural Systems*, 2000.

[49] S. Douglas and M. Gupta, *Convolutive Blind Source Separation for Audio Signals*, ser. Signals and Communication Technology. Springer Netherlands, 2007, pp. 3–45.

[50] S. C. Douglas, M. Gupta, H. Sawada, and S. Makino, "Spatio–temporal fastica algorithms for the blind separation of convolutive mixtures," *IEEE Transactions on Audio, Speech and Language Processing*, 2007.

[51] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal Processing*, 2001.

[52] Y. Li, S. Amari, A. Cichocki, D. W. C. Ho, and S. Xie, "Undercomplete blind subspace deconvolution based on sparse representation," *IEEE Transactions on Signal Processing*, 2006.

[53] Z. Szabó, B. Póczos, and A. Lőrincz, "A.: Undercomplete blind subspace deconvolution via linear prediction," Tech. Rep., 2007.

[54] S. Doclo and M. Moonen, "Gsvd-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Transactions on Signal Processing*, 2002.

[55] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New insights into the noise reduction wiener filter," *IEEE Transactions on Audio, Speech and Language Processing*, 2006.