Diss. ETH No 11052

On Nonlinear Filtering for Noise Reduction Using a Sensor Array

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY
ZURICH

for the degree of
Doctor of Technical Sciences

presented by

WOLFGANG KNECHT
Diplom Physiker Philipps Univ. Marburg
M.S. Physics MIT
born January 10, 1962
citizen of Germany

accepted on the recommendation of

Prof. Dr. G.S. Moschytz, examiner
Prof. Dr. J.L. Massey, co-examiner
Dr. F. Bonzanigo, co-examiner

1995

I don't know what I may seem to the world, but, as to myself, I seem to have been only like a little boy playing on the sea shore, and diverting myself in now and then finding a smoother pebble or prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me.

Isaac Newton shortly before his death in 1727.

Seite Leer /
Blank leaf

Acknowledgments

My voyage on the "great ocean of truth" was supported generously by Prof. George Moschytz who provided an excellent research environment for me within the Signal and Information Processing Lab. He granted me the privilege of elaborating on an original topic with the maximum intellectual freedom.

I would like to thank Prof. Jim Massey and Dr. Federico Bonzanigo for serving as co-examiners.

During my years at the ETH, I discovered some "non-mathematical lemmas". One important lemma was that teachers can learn from their students. I profited from teaching courses and supervising Semester and Diploma theses of Moreno Giulieri, Tunç Kütükçüoglu, Rolf Steiner and Marcel Joho who all contributed significantly to this research.

I am also indebted to the "linear" multi-microphone people Pat Peterson, Pat Zurek, Julie Greenberg, Bill Rabinowitz and Lou Braida with whom I worked in the Research Lab of Electronics at MIT. Without their inspiration this work would not have been possible.

Thanks go the my colleagues at the ETH for their helpfulness in various scientific matters around this project. In particular, I want to express my gratitude to my friend and colleague Markus Schenkel. Equipped with our NIPS coffee cups, we discussed and analyzed not only our projects but also the entire world.

I am grateful to my parents who lovingly supported me in the last 33 years and to Patricia Greenky who provided articles on speech and hearing from the New York Times. Special thanks go to my wife Sharmon. During many hours we discussed and improved not only the English, but her insightful comments proved again that an expert can benefit from the lay man's different perspective. Our little daughter Moraya "Dufina" also contributed to this thesis in her own special way by filling my heart with a new magnificent love and giving me more inner strength than ever before.

Table of Contents

Zusammenfassung

Störgeräusche stellen eines der grössten Probleme von momentan erhältlichen Hörgeräten dar. In letzter Zeit ist der Einsatz von Mehrfach-Mikrophonsystemen zur Unterdrückung von Hintergrundgeräuschen populär geworden. Die vorliegende Arbeit beschäftigt sich mit einem solchen System mit zwei Mikrophonen. Das System beinhaltet einen "Adaptive Noise Canceller" mit einem nichtlinearen Filter. Bisher wurde der "Adaptive Noise Canceller" üblicherweise mit linearen Filtern betrieben, wobei die Filterkoeffizienten so eingestellt wurden, dass der mittlere quadratische Fehler des adaptiven Filters minimiert wurde. Ein nichtlineares Filter kann diesen Fehler im allgemeinen noch verringern und dadurch einen verbesserten Signal-Rauschabstand am Ausgang des Systems erreichen.

Optimale lineare und nichtlineare Filter werden für eine Rauschquelle mit verschiedenen Wahrscheinlichkeitsdichten berechnet. Weiterhin wird untersucht, ob das Volterrafilter und das Perceptron, zwei adaptive nichtlineare Filter, in der Lage sind, das optimale Filter zu approximieren. Hierbei sind die Konvergenzzeit und die Störgeräuschunterdrückung im eingeschwungenen Zustand wichtige Kriterien zur Beurteilung dieser Filter.

Das Volterrafilter wird ebenfalls zur Unterdrückung eines zweiten Sprechers eingesetzt. Zwar minimiert das Filter seinen mittleren quadratischen Fehler, das Ziel ist aber, die Verständlichkeit des gewünschten Sprechers zu erhöhen. Die verarbeiteten Signale werden deshalb mit dem "intelligibility-weighted gain" beurteilt. Die Berechnung dieses Masses benötigt den Signal-Rauschabstand am Ausgang des Systems. Im Fall des nichtlinearen Systems kann dieser nur bestimmt werden, wenn sich keine Nutzsignalkomponenten im Referenzkanal des "Noise Cancellers" befinden. In diesem Idealfall ergibt sich, dass das quadratische Volterrafilter das Verständlichkeitsmass um maximal 2 dB gegenüber einem linearen Filter verbessern kann.

Seite Leer /
Blank leaf

Abstract

Background noise is one of the major problems of currently available hearing aids. Array processing techniques have become a popular research topic for reducing background noise. This work investigates a two-microphone beamformer which incorporates an adaptive noise canceller with a nonlinear filter. In adaptive noise cancelling, linear filters have been used to minimize the mean squared difference between the filter output and the desired signal. Depending on the probability densities of the involved signals, however, nonlinear filters can further reduce the mean squared difference, thereby improving signal-to-noise ratio at the noise canceller output.

In the case of a single noise source emitting an i.i.d. random process, optimum linear and nonlinear performance limits are established for various noise probability densities. To approximate optimum performance, two nonlinear adaptive architectures are realized, the Volterra filter and the multi-layer perceptron. Convergence speed and steady state performance are scrutinized.

The Volterra filter is also examined for speech interference. The beamformer is adapted to minimize the mean squared difference, but performance is measured with the intelligibility-weighted gain. This criterion requires the signal-to-noise ratio at the beamformer output. For the nonlinear processor, this can only be determined when no target components exist in the reference channel of the noise canceller so that the target is transmitted without distortion. Under these ideal conditions and at equal filter lengths, the quadratic Volterra filter improves the intelligibility-weighted gain by maximally 2 dB relative to the linear filter.

Chapter 1: Introduction

Hearing aid users frequently encounter the problem of background noise reducing the intelligibility of a desired talker. Recent studies reveal that this is the major source of dissatisfaction [1] and that over 90% of test subjects are discontent with their devices in situations with background noise [2]. In 1993, the American Food and Drug Administration (FDA) issued a warning to hearing-aid producers in regards to misleading advertising [3]. Several companies had claimed that their products were "able to distinguish speech sounds from all undesired noises". According to the FDA, these claims had been greatly exaggerated. In short, the background noise problem in hearing aids still awaits a feasible solution.

1.1. Noise Reduction for Hearing Aids

Currently, most commercially available hearing aids are equipped with a single microphone. To distinguish a desired signal from noise, single-microphone noise suppression systems exploit signal properties such as short-time stationarity [4] or periodicity [5]. Although these techniques increase broadband signal-to-noise ratio, they generally cannot improve speech intelligibility [6, 7, 8].

In real environments, interfering sounds often arrive from other directions than that of the signal of interest. The "directionality" of incoming signals is another important feature to distinguish the signal of interest from noise. Most conveniently, this feature is (ideally) independent of the properties of the involved signals[1]. For example, if a system is based on signal properties

[1] Chapter 3 shows that the Griffiths-Jim noise reduction system, for example, operates independently of the signal spectral densities under ideal conditions. Under more realistic conditions, however, performance does depend on the spectra.

alone it will be inherently difficult to separate two speech signals [9]. Using directional information, this task can be greatly facilitated. To take advantage of directionality, a noise reduction system requires at least two microphones. Today's most effective techniques for improving speech intelligibility in noise are indeed *microphone arrays* [10, 11, 12, 13]. The newest products on the market are now equipped with two microphones for noise reduction [14].

1.2. Why Nonlinear Processing?

Two developments have elicited the signal processing community's interest in nonlinear signal processing techniques in recent years. First, the increasing popularity of artificial neural networks has encouraged this trend. Second, rapidly advancing computer technology, especially microprocessor technology, has facilitated the implementation of complex algorithms for nonlinear systems. It is interesting to note that CPU performance of microprocessors almost doubled each year in the last decade [15].

Using Bayes "minimum mean squared error" (MMSE) estimation theory, we now describe why nonlinear processing is expedient for enhancing noisy signals. The performance gain over linear filtering depends on the statistical characteristics of the involved signals. In at least one important case, however, nonlinear filtering will not improve performance compared to standard linear filtering. This exception is discussed below.

Suppose we measure a random data vector

$$\boldsymbol{X}(k) = (X(k)\, X(k-1) \cdots X(k-N))^T \qquad (1.1)$$

at time k. The data vector $\boldsymbol{X}(k)$ consists of successive components of the stochastic process $X(.)$. When $\boldsymbol{X}(k)$ is regarded as the input vector of a tapped delay line or transversal filter, the integer N is called the filter length. The task is to find an estimate $\hat{S}(k)$ of a random variable $S(k)$ based on the data vector

$\boldsymbol{X}(k)$ such that the 'mean squared error' (MSE)

$$E[(S(k) - \hat{S}(k))^2] \tag{1.2}$$

is minimized. The symbol $E[.]$ denotes the expectation operator for ensemble averaging. Except for time-invariant beamforming in Chapter 3, time-adaptive linear and nonlinear processing in this study is based exclusively on the MSE performance criterion. To simplify notation, the time argument k is omitted in the following discussion. If the conditional probability density function $p_{S|\mathbf{X}}(.|\boldsymbol{x})$ is known, the optimum (Bayes) filter estimates S from a given data vector $\boldsymbol{X} = \boldsymbol{x}$ as the conditional mean

$$\hat{s}_B(\boldsymbol{x}) = E[S|\boldsymbol{X} = \boldsymbol{x}] = \int_{-\infty}^{+\infty} s\, p_{S|\mathbf{X}}(s|\boldsymbol{x})\, ds. \tag{1.3}$$

The Bayes estimator yields the minimum mean squared error (MMSE) given by

$$MMSE = \int_{-\infty}^{+\infty} ds \int_{R^{N+1}} d\boldsymbol{x}\, (s - \hat{s}_B(\boldsymbol{x}))^2\, p_{S,\mathbf{X}}(s, \boldsymbol{x}), \tag{1.4}$$

where $p_{S,\mathbf{X}}(.,.)$ is the joint probability density function. In general, the Bayes estimator is a nonlinear function of the input data vector $\boldsymbol{x}$. But if $\boldsymbol{X}$ and S are *jointly Gaussian*, the Bayes estimator (1.3) is a linear function of the data vector and identical to the finite impulse response Wiener filter [16].

The following example illustrates the difference between linear and nonlinear filtering for noise reduction. Suppose we measure the noisy signal

$$X(.) = S(.) + N_a(.), \tag{1.5}$$

where $S(.)$ is the signal of interest and $N_a(.)$ is an additive noise independent of $S(.)$. It is assumed that $S(k)$ is Gaussian-distributed with zero mean and variance $\sigma^2 = 0.5$ for all k. Let

the noise signal be Laplacian-distributed with the probability density function $p_{N_a}(.) = \frac{1}{2}\, e^{-|.|}$. According to the Wiener-Hopf equation, the optimum linear memoryless ($N = 0$) filter has a single weight given by

$$c_{opt} = \frac{E[S(k)\,X(k)]}{E[X(k)^2]} = \frac{\sigma_S^2}{\sigma_S^2 + \sigma_{N_a}^2} = \frac{1}{5}, \tag{1.6}$$

where stationary target and noise signals are assumed. Hence, the optimum linear estimator is

$$\hat{s}_L(x) = c_{opt}\, x = \frac{1}{5}\, x. \tag{1.7}$$

In Appendix B, we show that the optimum memory-less nonlinear filter is

$$\hat{s}_B(x) = \frac{1}{2}\, \frac{e^{x+\frac{1}{4}}\,(erf(x+\frac{1}{2}) - 1) + e^{-x+\frac{1}{4}}\,(erf(x-\frac{1}{2}) + 1)}{e^{x+\frac{1}{4}}\,(-erf(x+\frac{1}{2}) + 1) + e^{-x+\frac{1}{4}}\,(erf(x-\frac{1}{2}) + 1)}, \tag{1.8}$$

where $erf(.)$ is the error function defined as

$$erf(x) = 2/\sqrt{\pi} \int_0^x exp(-t^2)\, dt.$$

Figure 1.1 plots both the linear and nonlinear estimators as a function of the measured input x. To determine the performance of the Bayes estimator, the integral in (1.4) must be evaluated. In most cases this will be an intractable task. Even in our simple example with Laplacian noise, we did not succeed in evaluating the integral exactly. Performance can be estimated, however, by implementing the Bayes estimator, filtering a sufficiently long signal $x(.)$ and measuring the MSE at the output. Filtering 80,000 samples with the estimators (1.7) and (1.8) produces MSEs of 0.4014 and 0.3808, respectively. The same experiment with uniformly-distributed noise of equal mean and variance yields an MSE of 0.3715 for the nonlinear estimator. In the linear case, the performance of the optimum filter can be calculated easily. It is

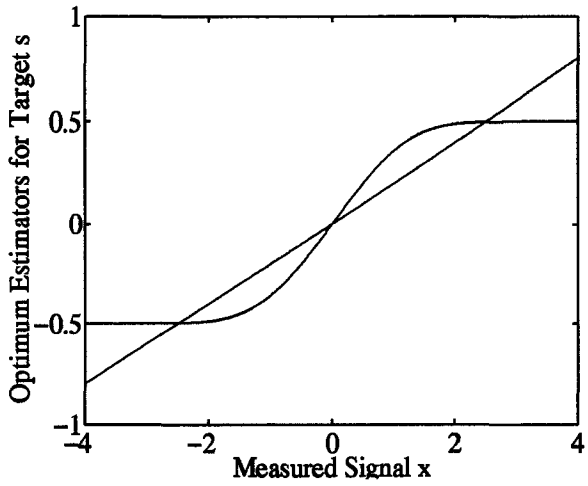


Figure 1.1: Optimum linear and nonlinear estimators for a Gaussian target in Laplacian noise.

MSE $= E[S^2] - E[\hat{S}_L^2] = 0.4$, independent of the noise probability density provided that the mean and variance of the noise do not change. This example and other experiments in Chapter 5 indicate that the more the shape of the noise probability density deviates from the bell-shaped Gaussian curve, the greater the improvement provided by nonlinear processing for constant filter length N.

Before nonlinear filters may be implemented in a hearing aid, it is important to ascertain whether typical environmental noises are sufficiently non-Gaussian. In many situations, acoustic interference is speech or music, e.g. in stores, offices or restaurants. The *long-term* amplitude density function of speech is well approximated by Laplacian or Gamma densities [17, 18], where speech is assumed to be stationary. Consequently, one can conclude that time-invariant nonlinear processing will perform better than linear processing for constant filter length. On the other hand, speech can be regarded as a non-stationary signal containing successive quasi-stationary voiced and unvoiced sounds. The voiced sounds comprise periodic components which usually have non-Gaussian amplitude distributions. Figure 1.2 depicts the amplitude distribution of a vowel "a", sampled at 8 kHz and 16 bits

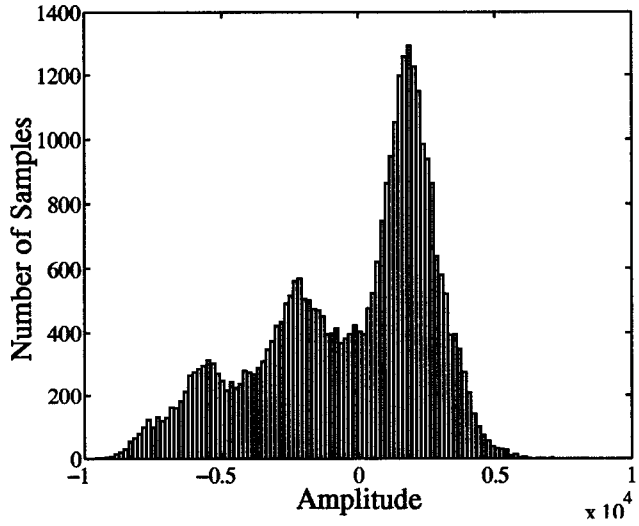


Figure 1.2: Typical asymmetric amplitude distribution for the vowel "a".

for 34,000 samples. Similar non-Gaussian shapes were found for other recordings of vowels. The short-term quasi-stationary structure of speech interference suggests an *adaptive* nonlinear filter for its attenuation. Because the amplitude distributions of voiced sounds are generally "more non-Gaussian" than the Laplace or Gamma distributions, it is expected that the improvement of nonlinear filtering relative to linear filtering will be more apparent when adaptive rather than time-invariant processing is carried out (see Chapter 5 for an experimental verification).

Typical environments contain a number of independent noise sources. If the interference is a superposition of independent stochastic processes, its amplitude distribution may be asymptotically Gaussian according to the central limit theorem (CLT) and linear filtering would be optimum[2]. The Lindberg condition [19], i.e. the individual variances of each process at a fixed time are small compared to the sum of these variances, is the neces-

[2]Linear filtering is optimum when noise *and* desired signal are jointly Gaussian. This comment refers to the ideal Griffiths-Jim beamformer in Chapter 3 where target estimation is independent of the target statistics. If the interference alone is Gaussian, then the optimum filter in this noise suppression system is linear.

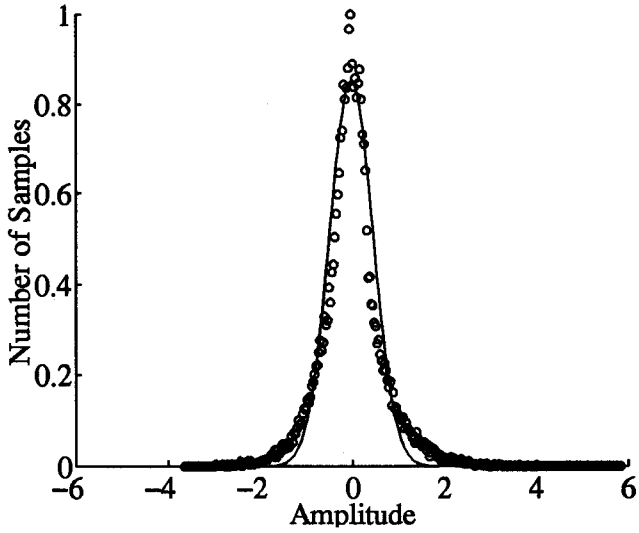


Figure 1.3: Amplitude distribution of a sum of ten independent talkers (circles) and a fitted Gaussian curve (solid line).

sary and sufficient condition for the CLT to hold. This condition is satisfied, for example, when all components of the sum have the same variance. Webster [20] argues that this situation rarely occurs in a natural environment. One would rather expect a small number of very strong sources and a large number of weak sources. He shows that the infinite sums $\sum X_k/k$ of i.i.d. random variables X_k have non-Gaussian probability densities when the distribution of X_k is Laplacian or sinusoidal. The same holds for sums $\sum X_k/2^k$.

As an example for real signals, consider Figure 1.3 depicting the amplitude distribution of the sum of ten independent speech signals with equal level. Each component consisted of a different sentence spoken by a female (five times) or a male (five times). The signals were sampled at 8 kHz and 16 bits for 30,000 samples. The measured data (circles) were fitted by a Gaussian (solid line) with unit area. Observe that the tails of the measured distribution decrease slower than those of the Gaussian. A similar graph is obtained when the components of the sum are weighted by a factor $1/k$ for $k = 1...10$. To summarize, one cannot always assume Gaussian statistics of a sum of independent noise signals.

1.3. Goals of This Work

This thesis investigates the two-microphone Griffiths-Jim (GJ) beamformer [21], one of today's most promising *adaptive* noise reduction devices, using a *nonlinear* filter in its reference channel. While performance limits for the GJ beamformer with a linear filter have been established [22], this study attempts to find these limits for the first time in the nonlinear case. Calculating optimum nonlinear filters according to (1.3) was found to be difficult, so only relatively simple situations with one i.i.d. noise process are considered.

In realistic situations, the required conditional probability densities for (1.3) are generally not available. Two nonlinear filter realizations, the Volterra filter and the perceptron, are used to approximate the unknown optimum Bayes filter function. It is particularly interesting to examine these nonlinear filters when the optimum Bayes filter is known and serves as an absolute benchmark. When the optimum nonlinear filter is unknown, this work adapts linear and nonlinear filters under the same conditions and compares their performance. Finally, it is of vital importance to estimate the improvement of speech intelligibility by nonlinear processing. Under ideal conditions, it is possible to employ the "intelligibility-weighted gain" described in Chapter 5.

Linear adaptive FIR filtering with the MSE criterion is a practical signal processing method. The extension to the nonlinear realm generally results in smaller MSEs. But simultaneously, one has to deal with much more complex mathematics and (still) open questions about the actual realization of these systems. This work primarily evaluates the potential benefits of nonlinear processing techniques for beamformers independently of implementation issues based on current technology. Although this thesis presents theoretical and experimental results, it cannot claim to offer a complete analysis of the proposed concepts. It is rather a first step towards a comprehensive assessment of nonlinear filters in multiple-sensor noise-reduction systems.

Chapter 2: Previous Research

This chapter reviews salient research in the field of noise suppression and evaluates these concepts with respect to hearing-aid applications. It is difficult to compare noise-reduction techniques because they are usually tested under different conditions with different performance measures. However, it is possible to identify advantages and disadvantages of each technique when applied to hearing-aids. The first part discusses approaches based on single-sensor recordings of noisy speech. These techniques are based on the additive-noise model (1.5), i.e., only the composite signal $X(.) = S(.) + N_a(.)$ measured by a single microphone is available. The second part describes systems using multiple sensors. These systems process the inputs of several microphones simultaneously.

2.1. Single Microphone Approaches

The periodic structure of voiced speech sounds can be used to remove unwanted additive noise from the measured signal $X(.)$. Through determination of the fundamental frequency (pitch), the harmonics of voiced speech can be identified and extracted. Simultaneously, frequency components of the noise outside of the "harmonic comb" are suppressed. The *comb filter* has been realized in the time-domain [23] and in the frequency-domain [5, 24, 8]. The method strongly relies on correct pitch information. Measuring the pitch of a noise-corrupted speech signal has been shown to be difficult, especially in the case of interfering speech [24]. Even if correct pitch information (obtained from clean speech) is supplied to the comb filter, time-domain processing *decreased* speech intelligibility in white noise despite an increase of output signal-to-noise ratio (SNR). In the case of interfering multi-talker speech babble, Kates [8] reported similar

results. For one interfering talker, Stubbs and Summerfield [24] obtained a slight intelligibility improvement, but pitch determination was not based entirely on the corrupted input signal.

Another approach is short-time *Wiener filtering.* For example, Graupe's Zeta noise blocker [25] is a modified adaptive short-time Wiener filter. The filter control mechanism requires noise characteristics which change more slowly than those of speech. Van Tasell at al. [26] tested this system and could not measure an intelligibility improvement. Ephraim suggested a time-varying Wiener filter driven by hidden Markov models for speech and noise [7]. He reported SNR improvements, but no intelligibility tests were provided.

Both comb filtering and Wiener filtering usually increase broadband SNR. *Speech intelligibility, however, depends primarily on SNR in third-octave bands according to articulation theory* [27]. The above methods (or any other linear adaptive filter) cannot change the within-band SNR. Hence, no intelligibility improvement is expected except in cases where the noise is confined to a frequency region which does not contain important speech information [28]. Experimental evaluations of the above filtering techniques confirmed these expectations. Nevertheless, linear adaptive filtering may still be appropriate for the improvement of listening comfort with a given intelligibility.

Noise suppression by *spectral subtraction* assumes that stationary intervals of the noise are longer than stationary intervals of speech [4]. The noise spectrum is measured during speech pauses and subsequently subtracted from the spectrum of the noisy speech. When the subtraction results in negative spectrum amplitudes for the speech estimate, the amplitude is set to a small positive value. This results in the famous "musical noise" inherent in the processed speech. Lim and Oppenheim investigated spectral subtraction with respect to intelligibility and could not find an improvement [29]. While Lim and Oppenheim presented the processed speech directly to the human listener, Hirsch used spectral subtraction as a *preprocessor for cochlear implants* and

measured intelligibility improvements [30]. Furthermore, spectral subtraction increased the recognition rate of hidden-Markov-model-based speech recognition in adverse environments [31].

Recently, Hardwick et al. [32] introduced a model-based speech enhancement system. Voiced and unvoiced components of the noisy speech are processed simultaneously. The system requires a pitch estimate and knowledge of the noise power density which is usually not available. For white noise with known variance, the system has been shown to increase intelligibility by at least 15 % of recognized words.

2.2. Multiple-Microphone Approaches

One of the earliest multi-microphone noise-suppression systems was suggested by Kaiser and David in [33, 34]. The system was equipped with two microphones. The sum signal of the two sensors was modulated by a gating signal which was derived from the cross-correlation between the two inputs. For a target source which was equidistant from the two sensors, the cross-correlation was low during speech pauses. In these intervals, the gating signal was increased, the sum signal was attenuated and the noise power was reduced. However, McConnell examined this technique in listening tests and could not find increased recognition rates [35]. This can be explained by the same argument used for several single-microphone techniques: the SNR in individual frequency bands of the sum signal was not changed by the gating.

Widrow et al. [36] introduced the principle of adaptive noise cancelling with two microphones in 1975. One microphone picked up the desired signal plus the ambient noise (primary signal) while the second microphone (located away from the desired signal source as far as possible) recorded only the noise field (reference signal). Assuming that the desired signal and the noise are uncorrelated and that the noise signals at the two microphones are correlated, an adaptive filter transformed the reference signal into the noise component of the primary signal con-

taining also the desired signal. Subtracting the filter output from the primary signal resulted in a substantial SNR gains. Unfortunately, in a hearing-aid application it is not possible to place a second microphone close to the source of interference. The adaptive beamformer in Section 3.4 uses an adaptive noise canceller but circumvents this problem by preprocessing the microphone signals.

In 1981, Strube [37] presented a two-microphone speech enhancement method whose principle is identical to that of the adaptive GJ beamformer introduced in detail in Section 3.4. Strube reported objective SNR improvements as well as increased single-word intelligibility of the processed speech in an anechoic environment. The success of this method has been reconfirmed by various investigators for different test material and acoustic environments [38, 39, 40, 41, 42, 12]. Two studies [41, 12] evaluated its performance with criteria based on articulation theory. The performance gain was determined relative to a single-microphone signal. The reported benefits through adaptive beamforming are expected to be smaller, however, when the gain is calculated with respect to the *sum* of the microphones (see the discussion of the delay and sum beamformer in Section 3.3.).

Since the adaptive beamformer's performance degrades in resonant environments, time-invariant beamformers have been suggested in [10, 11, 43]. These systems are not prone to target cancellation and are easier to implement than adaptive beamformers. In order to achieve a sufficient intelligibility improvement, more than two microphones are usually required. For a more detailed description of time-invariant and time-adaptive beamforming techniques, the reader is referred to Chapter 3.

Relatively little research has been conducted on *nonlinear* processing schemes for microphone arrays. Mitchell *et al.* [44] examined a four-microphone system with two nonlinear processing stages. It was shown that a combination of positive and negative full-wave rectifiers in the two stages can eliminate impulsive noise. For white noise or speech interference however, sim-

ple linear delay and sum beamforming achieved a higher output broadband SNR than the nonlinear method. No intelligibility measurements were provided. Souloumiac et al. proposed a GJ beamformer with a nonlinear memoryless Volterra filter for the rejection of narrowband non-Gaussian interference [45]. Simultaneously and independently, the same method was applied to broadband non-Gaussian noise using a Volterra filter with memory [46]. Nonlinear filtering resulted in increased jammer power rejection relative to linear processing for constant filter lengths. Listening tests were not performed. The work in [46] and a follow-up [47] are parts of this thesis.

The author is aware of only one study of nonlinear beamforming where a listening test has been conducted [48]. A *time-invariant* minimum MSE beamformer with two microphones was implemented with optimum linear and third-order Volterra tapped delay lines of length two. For speech babble interference, nonlinear processing exhibited a small improvement in intelligibility relative to linear processing - at the expense of a slightly distorted target signal at the output of the nonlinear beamformer. These results encouraged the closely related research presented in this thesis. As opposed to time-invariant beamforming in [48], we investigate in this work time-adaptive GJ beamforming with a nonlinear filter. This system has the ability to change its directivity pattern when the ambient noise field changes. It is examined in Chapter 5.

2.3. Summary of Chapter 2

- The majority of single-microphone noise suppression techniques is not well suited for improving intelligibility in hearing-aids. However, single-microphone enhancement of noisy speech is still useful as a preprocessing for speech recognizers or cochlear implants.
- Microphone arrays proved to be more successful in terms of improving intelligibility. Time-invariant or time-adaptive systems or a hybrid of both are currently the most promising candidates for a hearing-aid application.
- Nonlinear processing of noisy speech signals has been explored only marginally yet.

Chapter 3: Array Processing

Array processing or beamforming provides a method of rejecting unwanted spatial interference and simultaneously emphasizing the strength of a signal from a desired direction. To distinguish between interference and the desired signal, beamforming exploits phase and intensity differences measured at a sensor array. Since the late 1950's, this technique has been applied mainly to problems in the fields of radar, sonar and seismic signal processing but also to speech enhancement. The reader is referred to [49, 50] for excellent introductions to beamforming and its applications.

This chapter elaborates some fundamental facts about linear equi-spaced sensor arrays. After presenting the most important time-invariant beamformers, we introduce the Griffiths-Jim (GJ) adaptive beamformer with a linear filter.

3.1. Perfect Separation of Signals

How many sensors are required to separate perfectly the signals emitted by a number of sound sources? The number depends on the acoustic environment, the geometrical distribution of sound sources and sensors in space, and the frequencies of the incoming signals. Even if the transfer functions from sources to sensors are not known, it is still possible to specify a necessary condition for perfect separation. This section shows that the number of sensors must be at least as great as the number of sound sources.

Assume n different point sources emitting signals $S_j(f)$ and m omni-directional sensors receiving signals $X_i(f)$, where all signals are represented in the frequency domain. Let the transfer function from source j to sensor i be denoted by $H_{ij}(f)$. The sensor signals

can be expressed as follows:

$$\begin{pmatrix} X_1(f) \\ \vdots \\ X_m(f) \end{pmatrix} = \begin{pmatrix} H_{11}(f) & \cdots & H_{1n}(f) \\ \vdots & & \vdots \\ H_{m1}(f) & \cdots & H_{mn}(f) \end{pmatrix} \begin{pmatrix} S_1(f) \\ \vdots \\ S_n(f) \end{pmatrix},$$

or equivalently

$$\boldsymbol{X}(f) = \boldsymbol{H}(f)\, \boldsymbol{S}(f). \tag{3.1}$$

To simplify notation, the argument f will be omitted in the following discussion. *If the sensor signal vector* $\boldsymbol{X}$ *and the transfer function matrix* $\boldsymbol{H}$ *are known, then (3.1) has a unique solution* $\boldsymbol{S} = \boldsymbol{S}^*$ *if and only if* $\boldsymbol{X}$ *lies in the range of* $\boldsymbol{H}$ *and* $rank(\boldsymbol{H}) = n$. To see this, recall that the product $\boldsymbol{HS}$ is a linear combination of the columns of $\boldsymbol{H}$ with weights equal to the corresponding components of $\boldsymbol{S}$. A solution to (3.1) exists if and only if the vector $\boldsymbol{X}$ lies in the space spanned by the columns of $\boldsymbol{H}$, in other words "$\boldsymbol{X}$ lies in the range of $\boldsymbol{H}$". Additionally, if the columns of $\boldsymbol{H}$ are linearly independent, meaning "rank $(\boldsymbol{H}) = n$', then the solution is unique. We can conclude:

- If there are fewer sensors than sound sources ($m < n$), then $rank\,(\boldsymbol{H}) < n$ and consequently, the signals cannot be retrieved perfectly.

- If there are at least as many sensors as sound sources ($m \geq n$), then the signals can be separated perfectly if the uniqueness conditions are satisfied.

Certain constellations of sources and sensors will not permit signal separation even when the number of sensors is not smaller than the number of sources. Consider the example in Figure 3.1. Signals S_1 and S_2 arrive simultaneously at both sensor M_1 and at sensor M_2. Clearly, both signals have identical phase differences between the two sensors. The transfer function matrix in an anechoic free space for this example is

$$\boldsymbol{H} = \begin{pmatrix} exp(-j\,2\pi f \tau_1) & exp(-j\,2\pi f \tau_1) \\ exp(-j\,2\pi f \tau_2) & exp(-j\,2\pi f \tau_2) \end{pmatrix}, \tag{3.2}$$

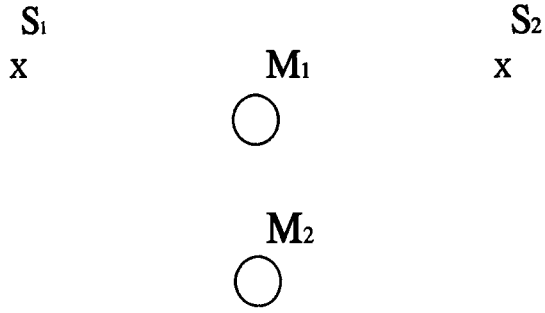


Figure 3.1: A separation of S_1 and S_2 is not possible.

where the τ_i are the times required by the signals to propagate from the sources to the sensors. Matrix $\boldsymbol{H}$ in (3.2) has rank only 1 for all frequencies, making the separation of S_1 and S_2 impossible.

Yanagida et al. [51] performed several experiments on sound source separation using the model (3.1). Under the assumption that the locations of microphones and sources are fixed, the authors determined the transfer function matrix $\boldsymbol{H}$ and solved equation (3.1) by multiplying with the pseudo-inverse of $\boldsymbol{H}$. To reach a satisfactory separation, the number of microphones had to be substantially greater than the number of sources. Jutten and Herault [52] presented a different approach whereby the source-sensor transfer functions are not required to be known. If the original signals are independent, their algorithm can separate a linear combination of signals by minimizing the cross correlation at lag zero between nonlinear functions of the reconstructed signals. This minimization makes the reconstructed signals statistically independent. An analog VLSI implementation of the Jutten-Herault algorithm has been used to separate two speech signals with two microphones [53]. Further experiments are described in [54, 55]. Others have investigated a simplified form of the Jutten-Herault algorithm, where separation is accomplished by *decorrelating* the estimated signals rather than making them independent [56, 57]. However, this concept discards higher-order cross correlations and, consequently, the adaptive decorrelation process may converge to undesired states [57], impeding a successful signal recovery.

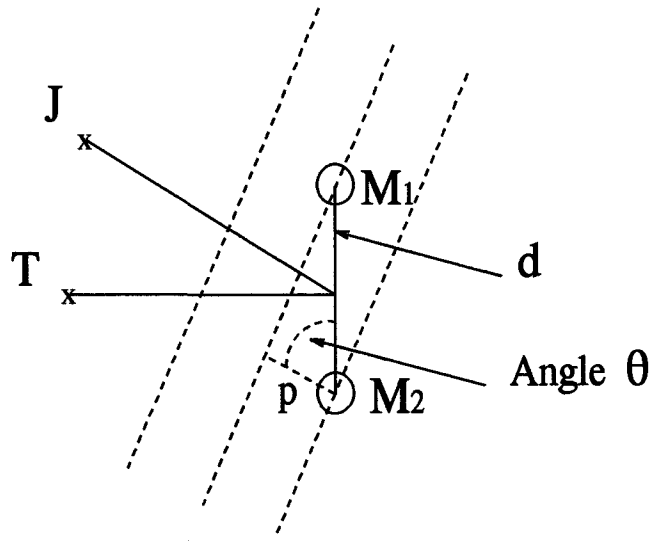


Figure 3.2: Situation with two microphones and two sources.

3.2. Spatial Aliasing

The frequencies of the incoming signals play an important role in an array's ability to distinguish the signals. This dependency is illustrated in the example for two sensors depicted in Figure 3.2. It is assumed that the inter-microphone distance d is sufficiently small compared to the distance between the array and the sources J and T so that the incoming wave fronts can be assumed to be planar. This assumption is adopted throughout this thesis. Consequently, the path difference between sensors for signal $J(.)$ is

$$p = d\,cos(\theta), \tag{3.3}$$

where θ denotes the angle of arrival, measured between the normal to the wave front and the line connecting the sensors. Signal $T(.)$ arrives simultaneously at both sensors, i. e., all frequencies are 'in phase' between M_1 and M_2. If signal $J(.)$ also contains in-phase frequencies, then the two signals are indistinguishable at these particular frequencies. This phenomenon is called *spatial aliasing*. Spatial aliasing occurs when the path difference p equals a multiple of the wave length λ of $J(.)$. The path difference p can be expressed as $p = c\,\Delta\,T_s$, where c is the velocity of sound, T_s denotes the sampling period and ΔT_s is the delay of $J(.)$ between

the two sensors measured in seconds. Using these definitions, the condition for spatial aliasing is

$$n\lambda = c\,\Delta\,T_s, \qquad n = 0, 1, 2, \cdots$$

This condition can be reformulated by employing the circular frequency ω :

$$\omega = \frac{n\,2\,\pi}{\Delta\,T_s}. \tag{3.4}$$

Equation (3.4) specifies the in-phase circular frequencies of $J(.)$ for the situation shown in Figure 3.2. As an example, suppose that the delay is $\Delta = 4$ samples. Spatial aliasing occurs at the normalized frequencies $\Omega = \omega T_s = 0, \frac{\pi}{2}, \pi, \cdots$. The transfer function matrix in (3.1) for this example is

$$\begin{aligned} \boldsymbol{H} &= \begin{pmatrix} exp(-j\,2\pi f\tau_1) & exp(-j\,2\pi f\tau_2) \\ exp(-j\,2\pi f\tau_1) & exp(-j\,2\pi f\tau_3) \end{pmatrix} \\ &= \begin{pmatrix} exp(-j\,2\pi f\tau_1) & exp(-j\,2\pi f\tau_2) \\ exp(-j\,2\pi f\tau_1) & exp(-j\,2\pi f\tau_2) \end{pmatrix}, \end{aligned}$$

where $H_{22} = exp(-j2\pi f(\tau_2 + 4T_s)) = exp(-j2\pi f\tau_2)\; exp(-j\Omega 4) = exp(-j\;2\pi f\tau_2)$ at the normalized frequencies $\Omega = 0, \frac{\pi}{2}, \pi, \cdots$. As expected, $\boldsymbol{H}$ is singular at the aliasing frequencies and non-singular at all other frequencies.

3.3. Time-Invariant Beamforming

In general, the transfer functions between sensors and sources are not known in array processing. These can be estimated from the microphone signals alone with adaptive signal separation algorithms [52, 56, 57, 58]. If one is interested in interference cancellation alone rather than in a complete recovery of all involved signals, it is not necessary to estimate the transfer function matrix. This substantially reduces the computational load on the array processor. Concentrating on the problem of interference cancellation, we discuss in this section the most important forms

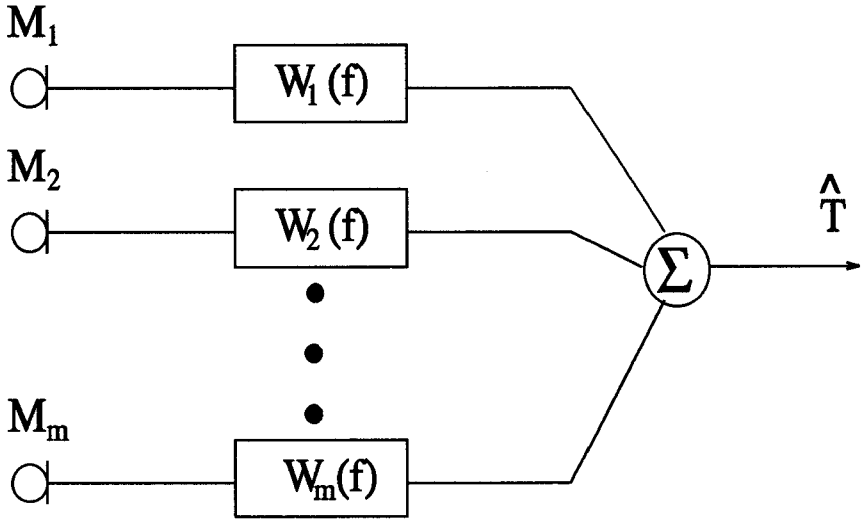


Figure 3.3: General structure of a beamfomer.

of time-invariant beamforming: delay and sum (DS) processing and maximum likelihood (ML) processing. Time-invariant beamforming is suitable for acoustic environments where interference arrives from many directions simultaneously. The ideal case of isotropic noise is discussed within the context of ML beamforming.

Consider the beamforming array in Figure 3.3 consisting of m microphones. Each microphone's signal is processed by a time-invariant filter with a transfer function $W_i(f)$, $i = 1 \ldots m$. The sum of the filter outputs provides an estimate of the target signal $\hat{T}$. Having defined a target direction, one can apply simple processing schemes to increase the target-to-jammer ratio (TJR) at the beamformer output relative to the TJR at a single sensor. Although the following discussion is restricted to linear arrays with equally spaced microphones, the theory is applicable to arbitrary array geometries. Two types of linear arrays are considered: the *endfire* array consisting of microphones along a line collinear with the target direction, and the *broadside* array consisting of microphones along a line perpendicular to the target direction. The broadside array receives an identical target signal at all sensors because the incoming sound waves are modelled as plane waves.

By employing appropriate time delays at the sensors, one can assure an identical target signal at the endfire sensors as well.

Let the beamformer input vector be $\boldsymbol{X}(f)$ according to (3.1) with $H_{ij} = exp(-j2\pi f\tau_{ij})$, where τ_{ij} is the *relative delay* of signal j at microphone i with respect to a reference microphone at the origin. Choosing microphone M_1 to be at the origin and an inter-microphone distance d, we find that the relative delays are

$$\tau_{ij} = \frac{d\,(i-1)\,cos(\theta)}{c}. \tag{3.5}$$

This relation may be verified easily in Figure 3.2 by adding additional microphones to the linear array. Note that the index j has disappeared on the right side of the last equation. The angle θ contains j only implicitly. The output of the system in Figure 3.3 is

$$\hat{T}(f) = \sum_{i=1}^{m} W_i(f)\,X_i(f) = \sum_{i=1}^{m} W_i(f)\,S(f)\,exp(-j2\pi f\tau_{ij}). \tag{3.6}$$

To obtain the *directional response* G_b of the beamformer in the direction specified by the angle of arrival θ, the output is divided by the signal $S(f)$ coming from that direction:

$$G_b(f,\theta) = \frac{\hat{T}(f)}{S(f)} = \sum_{i=1}^{m} W_i(f)\,exp(\frac{-j2\pi f d(i-1)\,cos(\theta)}{c}). \tag{3.7}$$

Equation (3.7) represents the directional response for a broadside array where the target direction is $\theta = \frac{\pi}{2}$. Note that for linear arrays the angle θ does not specify a direction but rather a rotationally symmetric cone on which the jammer source can have an arbitrary position. We will, however, still associate θ with a 'direction' in compliance with general 3-dimensional array theory. The endfire directional response is obtained by introducing steering delays $\tau_i^s = d(m-i)/c$ for the target direction specified by $\theta = 0$ so that microphone M_1 is the closest to the target source. Hence, the weights for the endfire array are

$W_i(f) = A_i(f)\,exp(-j2\pi f\tau_i^s)$, yielding the endfire directional response

$$G_e(f,\theta) = \sum_{i=1}^{m} A_i(f)\,exp(-j2\pi f\tau_i^s)\,exp(\frac{-j2\pi fd(i-1)\,cos(\theta)}{c}). \tag{3.8}$$

The broadside and endfire directional responses (3.7), (3.8) are expressed more compactly as an inner product

$$G(f,\theta) = \boldsymbol{W}^T\,\boldsymbol{E}, \tag{3.9}$$

where the weight vector $\boldsymbol{W}$ contains the components $W_i(f)$ and

$$\boldsymbol{E} = (1, exp(\frac{-j2\pi fd\,cos(\theta)}{c}), \cdots, exp(\frac{-j2\pi fd(m-1)\,cos(\theta)}{c}))^T. \tag{3.10}$$

Delay and Sum Beamforming

The simplest form of delay and sum (DS) beamforming incorporates delays at the sensors such that the target signal is identical at all sensors, and subsequently adds the equally weighted microphone signals. The broadside array does not require any delays whereas the endfire array is equipped with delays as shown above. As an example, consider $W_i(f) = A_i(f) = \frac{1}{6}$ for a six-element broadside and endfire system. Figures 3.4 and 3.5 show the magnitude of the directional responses G_e and G_b in dB versus angle θ (polar pattern) for $d = 2.8$ cm at $f = 4$ kHz and $f = 6.161$ kHz. For both geometries, the main lobe towards the target becomes narrower for the higher frequency. The attenuation in target direction is zero dB as expected. The broadside beamformer cannot attenuate signals at $\theta = 270°$ because these are in-phase at all sensors. On the other hand, it has a narrower main lobe than the endfire system. At exactly 6.161 kHz, the endfire array exhibits an undesired second lobe, called a *grating lobe*, at $\theta = 180°$ with zero dB attenuation. Whenever the distance d between two sensors is greater than half of the wavelength λ of the incoming signal, the endfire polar pattern will contain such

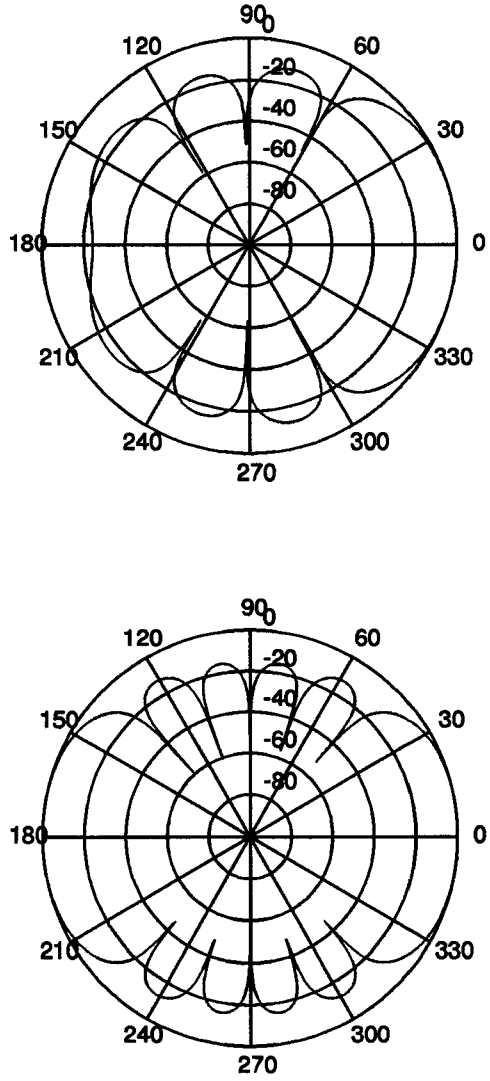


Figure 3.4: Polar pattern of 6 microphone endfire array at 4 kHz (top) and at 6.161 kHz (bottom).

grating lobes. To avoid grating lobes in the broadside polar pattern, the distance d must be smaller than λ. In the example, the first grating lobes for the broadside array appear at $f = 12.321$ kHz for $\theta = 0°$ and $\theta = 180°$. According to (3.3), the magnitude of the path difference p is maximal at these angles, causing spatial aliasing for $d = \lambda$. When the inter-microphone distance d is increased, the width of the main lobe becomes smaller, resulting in a higher spatial resolution. Simultaneously, more grating lobes appear which may be disastrous to performance if strong jammers arrive at those angles.

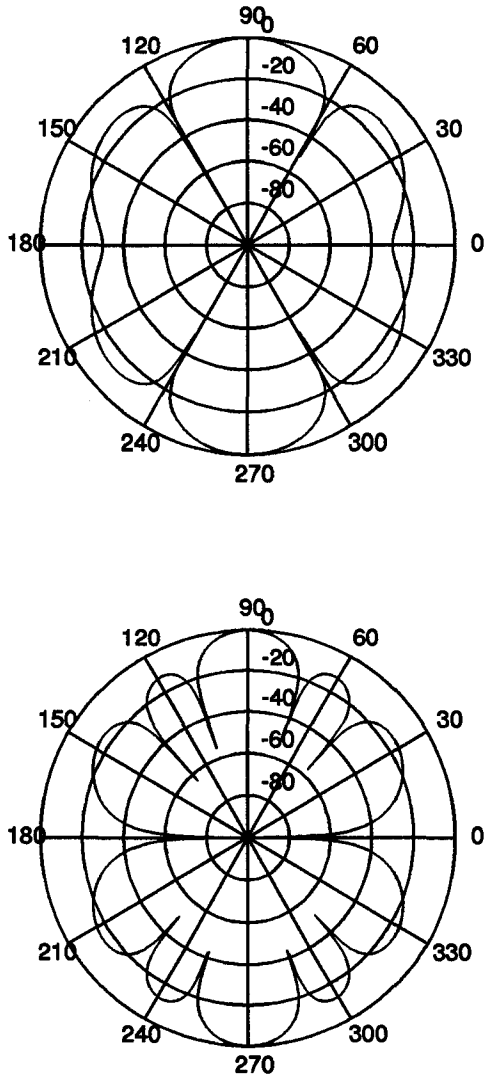


Figure 3.5: Polar pattern of 6 microphone broadside array at 4 kHz (top) and at 6.161 kHz (bottom).

Both amplitude weighting and phase correction at each sensor can be manipulated in various ways to obtain different shapes of the polar pattern. Soede provided a more detailed description of other frequency-independent amplitude weightings [59, 10]. The following section introduces the ML beamformer employing *frequency-dependent* amplitude weightings.

Maximum Likelihood Beamforming

This section describes a beamforming technique for optimizing an important performance criterion, namely the *directivity index*. The directivity index $D(f)$ is defined as the ratio of the directional response power in the target direction to the average directional response power from all other directions:

$$D(f) = \frac{\mid G(f, \theta_t, \phi_t) \mid^2}{\frac{1}{4\pi} \int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} \mid G(f, \theta, \phi) \mid^2 sin(\theta)\, d\theta\, d\phi}, \tag{3.11}$$

where the target direction is defined by the polar angles (θ_t, ϕ_t). Maximizing directivity is particularly well suited for *isotropic noise*, which is defined as the superposition of independent plane waves with identical spectra and uniformly distributed incident angles. It has been verified that the diffuse sound field composed of all reflected sounds in resonant environments is nearly isotropic [60]. Peterson [22] showed that the unbiased ML estimate of the target maximizes array directivity under the assumption of a Gaussian isotropic noise field. This estimator is also equal to the unbiased minimum variance estimator [22]. The m-dimensional weight vector of the ML beamformer is given by

$$\boldsymbol{W}^T(f) = \frac{\boldsymbol{E}(\theta_t, \phi_t)^\dagger\, \boldsymbol{S}^{-1}}{\boldsymbol{E}(\theta_t, \phi_t)^\dagger\, \boldsymbol{S}^{-1}\, \boldsymbol{E}(\theta_t, \phi_t)}, \tag{3.12}$$

where $\dagger$ denotes the complex conjugate transpose. The m-dimensional vector $\boldsymbol{E}(\theta_t, \phi_t)$, which was defined in (3.10), contains the free-field (no interfering objects) transfer functions from the target direction to the sensors in anechoic space (no reverberation). It is independent of the azimuthal angle ϕ for the linear arrays considered in this work. For the broadside array, for example, it is simply $\boldsymbol{E}(\theta = \frac{\pi}{2}) = (1,\ 1,\ \cdots,\ 1)^T$. Finally, the elements of the matrix $\boldsymbol{S}$ for isotropic noise are given by

$$S_{ij} = \Phi(f)\, \frac{1}{4\pi} \int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} E_i(\theta, \phi)\, E_j^*(\theta, \phi)\, sin(\theta)\, d\theta\, d\phi, \tag{3.13}$$

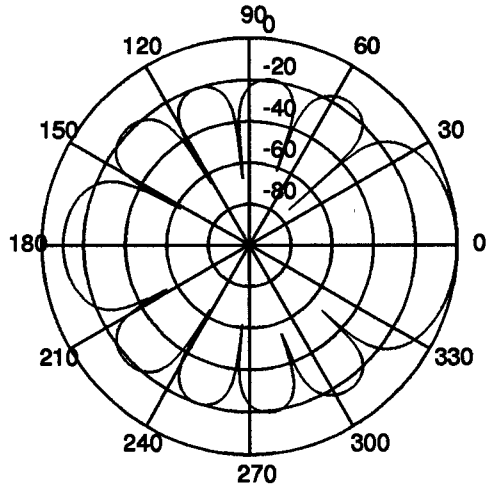


Figure 3.6: Polar pattern of ML beamformer for 6 microphone endfire array at 4 kHz.

where $*$ denotes the complex conjugate and $\Phi(f)$ is the common source spectral density. Note that $\Phi(f)$ cancels out in (3.12). Processor (3.12) will have unity gain in the target direction, consistent with the constraint of zero bias for the maximum likelihood estimator. It is worth pointing out that the weights (3.12) are optimal for arbitrary noise fields. A noise field is characterized by the matrix $\boldsymbol{S}$, which is the cross-spectral-density matrix for noise incident on the array. For example, the cross-spectral-density matrix for uncorrelated white sensor noise is $\sigma^2 I$, where σ^2 denotes the noise power at each sensor and I is the $m \times m$ identity matrix. Once $\boldsymbol{S}$ is known, the linear processor (3.12) minimizes noise output power under the constraint that the target signal direction has unity gain. Finally, it must be emphasized that (3.12) is also valid when the array is mounted on a head, but the target transfer function vector $\boldsymbol{E}(\theta_t, \phi_t)$ must then be modified to include head shadow effects.

Referring to the previous example with 6 microphones, Figure 3.6 depicts the polar pattern for the ML endfire array at 4 kHz. Compared to the corresponding DS polar pattern in the upper part of Figure 3.4, the six-element ML beamformer has a narrower main lobe and deeper side lobes (except at $\theta = 180°$),

	Max. Likelihood	Delay & Sum
1 kHz endfire	15.47	3.29
1 kHz broads.	5.52	1.04
2 kHz endfire	15.17	6.04
2 kHz broads.	5.70	3.29
4 kHz endfire	13.61	8.79
4 kHz broads.	6.43	6.04
6.161 kHz endfire	7.78	7.78
6.161 khz broads.	7.78	7.78
8 kHz endfire	7.38	7.20
8 kHz broads.	8.83	8.79

Table 3.1: Beamformer directivities in dB for several frequencies in isotropic noise. Microphone spacing is $d = 2.8$ cm and number of sensors is $m = 6$.

which improves directivity by about 5 dB. Directivities for frequencies between 1 kHz and 8 kHz are summarized in Table 3.1, which shows that optimum processing is especially advantageous at lower frequencies. DS directivity goes to zero for low frequencies or small sensor spacing ($d << \lambda$). ML directivity approaches *non-zero limits*, given by $10\log(m^2)$ and by approximately $10\log(\frac{4\lfloor(m-1)/2\rfloor+3}{\pi})$ for endfire and broadside arrays, respectively [22]. If frequency is high or sensor spacing is large ($d >> \lambda$) or d equals an integer multiple of $\lambda/2$, then both geometries have a directivity of $10\log(m)$. In this case, the isotropic noise is uncorrelated between sensors and the DS beamformer is identical to the ML beamformer. To exploit the advantage of ML endfire processing fully, one must ensure that $d << \lambda$. This can be achieved by either confining the total array span to a few centimeters (as required for a hearing aid) or by band-limiting the incoming signals to keep λ above a threshold. In contrast to the ML endfire array, the ML broadside processor does not reach its

maximum directivity for $d << \lambda$. Broadside arrays exhibit highest directivity ($10\log(m) < D < 10\log(m^2)$) when $d > \lambda/2$ [61]. For a head-sized array of several microphones, sensor distance is typically $d \approx 3$ cm. As a consequence, the condition $d > \lambda/2$ is only met for frequencies above 5.7 kHz so that maximum broadside performance cannot be fully exploited for hearing aids.

A real implementation of a beamforming device must take into account another important design criterion: robustness against noise that is uncorrelated between sensors (e.g. from microphone internal preamplifiers or tolerance errors). The measure "sensitivity" has been introduced to quantify this array property and is defined as the ratio of array output power due to uncorrelated sensor noise to the average sensor noise power at the microphones [62]. Array sensitivity increases monotonically for increasing directivity [43]. This may render the beamformer impractical, particularly for low frequencies or small sensor spacing. In other words, directivity is traded against sensitivity. It is possible to fix the array's sensitivity at a desired level and maximize directivity under that constraint [43].

3.4. Linear Griffiths-Jim Adaptive Beamforming

This section introduces the two-microphone Griffiths-Jim (GJ) beamformer [21] with a linear adaptive filter. GJ beamforming enhances a target signal arriving from a desired direction by suppressing simultaneously arriving jammer signals from other directions. In contrast to time-invariant beamforming, the GJ beamformer *adjusts* its adaptive filter so that the magnitude of the beamformer's transfer function in the jammer direction(s) is minimized subject to the constraint that the target transfer function is equal to one. One says that the beamformer "puts a null into the jammer direction(s)". If the direction of a jammer changes, the adaptive filter will track this movement by adjusting its weights, thereby putting a null into the new direction. The adaptive beamformer is more suitable for cancelling directional

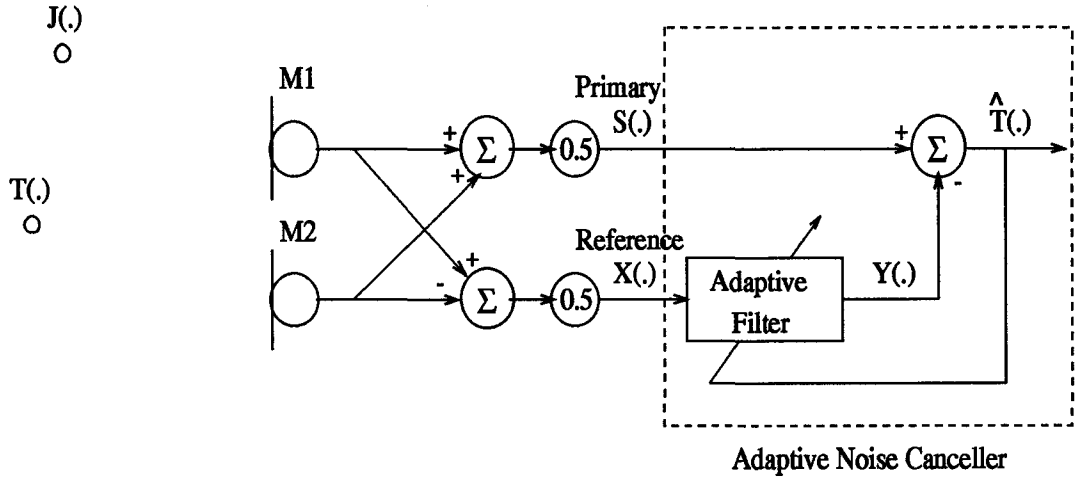


Figure 3.7: Two-microphone Griffiths-Jim beamformer for suppression of directional background noise.

interference than for reducing isotropic noise.

Figure 3.7 depicts the GJ beamformer with omni-directional microphones $M1$ and $M2$. The target source emitting the (speech) signal $T(.)$ is equidistant from the two microphones. An off-axis jammer signal $J(.)$ impinges on the microphones with a time delay of Δ samples between $M1$ and $M2$. For simplicity, we assume in this thesis that the jammer delay Δ is an integer multiple of the sampling period. Often signals already exist in sampled form for further digital processing. If non-integer arrival times are required, a digital low-pass filter can be used to interpolate between signal samples [63]. The scaled difference between the microphone signals

$$X(.) = \frac{1}{2}(J(.) - J(. - \Delta)) \tag{3.14}$$

contains no target components and is the reference input to the adaptive noise canceller[50]. The scaled sum of the signals

$$S(.) = T(.) + \frac{1}{2}(J(.) + J(. - \Delta)) \tag{3.15}$$

is the primary input to the noise canceller. Assuming that $T(.)$ and $J(.)$ are uncorrelated, the beamformer produces a target estimate $\hat{T}(.)$ by minimizing the output power $E[(S(k) - Y(k))^2]$

at time k, where $E[.]$ is the expectation operator and $Y(.)$ is the output of the adaptive filter. The signal $\hat{T}(.)$ is called the "minimum-variance distortionless estimate" of the target signal because ideally, the beamformer attenuates the interference without affecting the target. The system with an LMS-adapted linear FIR filter has been shown to improve target speech intelligibility in anechoic and weakly resonant environments containing one jamming source [38, 40, 42, 12, 13]. It must be emphasized that the beamformer produces an undistorted target estimate only when no target components exist in the reference channel. A misalignment of the target or a microphone gain mismatch will violate this condition and, consequently, the system will partially cancel the target signal. Exploiting the fact that the target is speech, the beamformer can be adapted only during speech pauses to avoid target cancellation [39, 41, 42]. Greenberg and Zurek [41] modified the GJ beamformer to alleviate the problem of target cancellation through misalignment. In [41], they worked with a delay equal to half the filter length in the primary channel. In [64], they recommended smaller delays making the beamformer more robust against target reverberation. Because in this thesis we emphasize the comparison between linear and nonlinear adaptive filters in the GJ beamformer, we restrict ourselves to the ideal case of a target-free reference channel.

Unconstrained Wiener Filter

The theoretical performance limits of the two-microphone GJ system will now be examined for the unconstrained Wiener filter. For this analysis, it is assumed that sound sources and microphones do not change their positions and that the jammer signals are stationary. More specifically, we suppose that the target $T(.)$ and jammers $J_1(.)$ and $J_2(.)$ are uncorrelated and that $J_1(.)$ and $J_2(.)$ are wide-sense stationary zero-mean stochastic processes. Note that Figure 3.7 shows only one jammer source.

Throughout this section, all calculations are performed in the z-transform domain. For simplicity, the argument of all z domain functions has been omitted except in those cases where it is z^{-1}.

The signal in the primary channel is

$$\begin{aligned} S &= H_T\, T + \frac{1}{2}\,[H_{11} + H_{21}]\, J_1 + \frac{1}{2}\,[H_{12} + H_{22}]\, J_2 \\ S &= H_T\, T + H_1^+\, J_1 + H_2^+\, J_2, \end{aligned} \tag{3.16}$$

and the signal in the reference channel is

$$\begin{aligned} X &= \frac{1}{2}\,[H_{11} - H_{21}]\, J_1 + \frac{1}{2}\,[H_{12} - H_{22}]\, J_2 \\ X &= H_1^-\, J_1 + H_2^-\, J_2, \end{aligned} \tag{3.17}$$

where the source-sensor transfer functions H_{ij} are defined as in Section 3.1. The target-sensor transfer function is denoted by H_T. The quantities H_i^+ and H_i^- in (3.16) and (3.17) are the weighted sums and differences of the H_{ij}, respectively. Here, primary and reference signals are given in a more general form than in (3.15) and (3.14). For the comparison between linear and nonlinear GJ beamformers, however, we use everywhere in this thesis the anechoic jammer transfer functions $H^+ = \frac{1}{2}(1 + z^{-\Delta})$ and $H^- = \frac{1}{2}(1 - z^{-\Delta})$ corresponding to (3.15) and (3.14). We chose these ideal transfer functions to simplify the calculation of optimum nonlinear filters and their performance.

Upon dividing the cross spectral density Φ_{XS} by the spectral density Φ_{XX}, one obtains the unconstrained Wiener filter for the GJ beamformer as

$$W^* = \frac{\Phi_{XS}}{\Phi_{XX}}. \tag{3.18}$$

The spectral densities can be derived as follows:

$$\Phi_{XX} = H_1^-\, H_1^-(z^{-1})\, \Phi_{J_1 J_1} + H_2^-\, H_2^-(z^{-1})\, \Phi_{J_2 J_2}. \tag{3.19}$$

$$\begin{aligned} \Phi_{XS} &= \Phi_{H_1^- J_1,\, H_1^+ J_1} + \Phi_{H_2^- J_2,\, H_2^+ J_2} \\ &= H_1^+\, \Phi_{H_1^- J_1,\, J_1} + H_2^+\, \Phi_{H_2^- J_2,\, J_2} \\ &= H_1^+\, \Phi_{J_1,\, H_1^- J_1}(z^{-1}) + H_2^+\, \Phi_{J_2,\, H_2^- J_2}(z^{-1}) \\ &= H_1^-(z^{-1})\, H_1^+\, \Phi_{J_1 J_1} + H_2^-(z^{-1})\, H_2^+\, \Phi_{J_2 J_2}. \end{aligned} \tag{3.20}$$

Inserting the power spectra from (3.19) and (3.20) into (3.18) leads to

$$W^* = \frac{H_1^-(z^{-1})\, H_1^+\, \Phi_{J_1J_1} + H_2^-(z^{-1})\, H_2^+\, \Phi_{J_2J_2}}{H_1^-\, H_1^-(z^{-1})\, \Phi_{J_1J_1} + H_2^-\, H_2^-(z^{-1})\, \Phi_{J_2J_2}}. \tag{3.21}$$

Inspection of the Wiener filter (3.21) reveals some interesting points. First, we note that the filter is independent of the target spectral density. Second, if one jammer J_i is identical to zero, we see that the filter simplifies to

$$W_1^* = \frac{H_j^+}{H_j^-}, \qquad i \neq j. \tag{3.22}$$

The Wiener solution is also independent of the jammer spectral density. The output of the beamformer becomes

$$\begin{aligned} S - Y &= H_T\, T + H_j^+\, J_j - W_1^*\, X \\ &= H_T\, T, \end{aligned} \tag{3.23}$$

that is to say, the beamformer can *completely* suppress the single jammer J_j regardless of its statistics and angle of arrival. The difference to time-invariant beamforming becomes apparent at this point. Recall from the previous section that endfire maximum-directivity beamforming with two sensors achieves at most an *average* jammer attenuation of $10\log(2^2) = 6$ dB, where the average is taken over all angles (θ, ϕ). Hence, the adaptive beamformer is much more efficient for directional interference.

It is worth noting that there is an adaptive structure similar to the one in Figure 3.7 which *separates* target and jammer perfectly. This can be accomplished by feeding a single microphone signal into the primary channel instead of the weighted sum of the microphone signals. Then, the Wiener filter transforms the reference signal into the room-filtered jammer signal. The GJ beamformer, however, has the advantage that, by building the weighted sum of the two microphone signals, the target-to-jammer ratio (TJR) increases relative to the TJR in each of the

microphone signals. This corresponds to the delay and sum (DS) beamformer discussed in Section 3.3. When microphone spacing is large compared to the incident wave length ($d >> \lambda$), the jammer signals between sensors are generally uncorrelated and as a result, the TJR in the primary channel will be $10\log(2) = 3$ dB higher than the TJR at each sensor.

Recalling our discussion of spatial aliasing, the reader may at first be surprised that, according to (3.23), the two-sensor beamformer can suppress a single jammer completely. What happens at the aliasing frequencies given in (3.4)? The Wiener filter (3.22) under anechoic conditions is

$$W_1^* = \frac{1 + z^{-\Delta}}{1 - z^{-\Delta}}. \tag{3.24}$$

This filter can be implemented in the time domain by the difference equation

$$Y(k) = Y(k - \Delta) + X(k) + X(k - \Delta) \tag{3.25}$$

with initial rest conditions $X(k) = 0$ and $Y(k) = 0$ for $k < 0$. Imagine a single off-axis interference with Δ samples delay between microphones and no target signal present. The initial rest conditions are met if $J(k) = 0$ for $k < 0$. According to (3.14), the filter input is $X(k) = \frac{1}{2}J(k)$ for $k = 0, 1, \cdots, (\Delta - 1)$ and $X(k) = \frac{1}{2}(J(k) - J(k - \Delta))$ for $k \geq \Delta$. It is easily verified that the filter output is exactly equal to the primary signal (3.15), i.e., the interference is suppressed completely. A sinusoidal jammer with an aliasing frequency given by (3.4) produces an input signal equal to zero *except at the onset* where $X(k) = \frac{1}{2}J(k)$ for $k = 0, 1, \cdots, (\Delta - 1)$. The corresponding output signal is identical to the primary signal, resulting in complete suppression of the jammer sinusoid. After the first Δ samples, this jammer becomes indistinguishable from a sinusoid arriving from the target direction. For the Wiener filter, the onset is sufficient to identify the jammer for all future values $k > 0$. As we will see in the next section, the optimum finite impulse response (FIR) filter cannot

cancel a jammer at the aliasing frequencies. Because of its finite memory, the FIR filter "forgets" the onset so that it cannot distinguish between the target and jammer after N samples, where N is the filter length.

Perfect cancellation of a single directional jammer is only possible under very ideal conditions that are usually not met in real acoustic environments. For example, if receiver noise is included in the calculations, it can be shown that jammer components appear at the beamformer output [50]. Another problem frequently encountered is that target components appear at the reference input. In this case, the target-to-jammer (TJR) density ratio at the beamformer output is reciprocal to the TJR density ratio at the reference input [50]. The Wiener filter is "distracted" from perfectly modeling the jammer in the primary channel and cancels the target instead.

Returning again to solution (3.21), let us analyze the situation in Figure 3.1 with two jammers, $S_1 = J_1$ and $S_2 = J_2$, plus an additional target from straight ahead (not shown). Since $H_{11} = H_{12}$ and $H_{21} = H_{22}$, one obtains $H_1^+ = H_2^+$ and $H_1^- = H_2^-$. In this case, the Wiener filter reduces to

$$W_2^* = \frac{H_1^+}{H_1^-} = \frac{H_2^+}{H_2^-}. \tag{3.26}$$

As in the example with a single jammer, the beamformer output is given by (3.23), i.e. both jammers are completely cancelled. Here, the difference between signal separation and signal cancellation becomes apparent. The two-microphone beamformer can suppress the two jammers perfectly but it cannot reconstruct the three signals T, J_1 and J_2 because the number of sources is higher than the number of sensors. If J_1 and J_2 arrive at the array with distinct phase differences between microphones, the beamformer output generally contains some residual jammer power, i.e., the two jammers cannot be cancelled completely by the two-microphone beamformer.

Optimum FIR Filter

The Wiener filter in the previous section is an infinite impulse response (IIR) filter. A changing acoustic environment requires an adaptation of the beamformer filter. Adaptive IIR filters are more difficult to realize than adaptive finite impulse response (FIR) filters. Adaptive IIR filters may become unstable during adaptation and their performance surfaces are generally non-quadratic [50]. Adaptive FIR filters do not suffer from these problems, which makes them more suitable for a real implementation. On the other hand, an optimum FIR beamformer will always reduce jammer power less than the optimum IIR filter. However, GJ beamformers with adaptive IIR filters have been investigated in [40, 42] and shown to provide practically no improvement over adaptive FIR filters.

This section examines the performance of the optimum FIR beamformer for arbitrary filter lengths N in the case of a single white jammer $J(.)$. The jammer delay between sensors is set to $\Delta = 1$. The target $T(.)$ is uncorrelated with $J(.)$. The samples of the reference signal (3.14) constitute the components of the input vector

$$\boldsymbol{X}(k) = \frac{1}{2} \begin{pmatrix} J(k) - J(k-1) \\ J(k-1) - J(k-2) \\ \vdots \\ J(k-N) - J(k-N-1) \end{pmatrix}. \tag{3.27}$$

Optimum linear FIR beamforming achieves the minimum output power [50]

$$MSE = E[S^2] - \boldsymbol{P}^T \boldsymbol{R}^{-1} \boldsymbol{P}, \tag{3.28}$$

where $\boldsymbol{P} = E[S(k)\boldsymbol{X}(k)]$ denotes the cross-correlation vector and $\boldsymbol{R} = E[\boldsymbol{X}(k)\boldsymbol{X}^T(k)]$ is the auto-correlation matrix. In Appendix B, it is shown that

$$MSE = \sigma_t^2 + \sigma_j^2 \frac{1}{N+2}, \tag{3.29}$$

where σ_t^2 and σ_j^2 designate the target and jammer variance, respectively. Without loss of essential generality, the target is assumed to be zero-mean and wide-sense stationary. This ensures constant target power in (3.29) but it is not a necessary requirement for the derivation of the MSE. Jammer attenuation is not influenced by the target in the ideal case of a target-free reference channel.

Several interesting observations can be made concerning (3.29). If $N = 0$, jammer power is reduced by half, producing a TJR improvement of $10\log(2) = 3$ dB. According to the Wiener-Hopf equation, the single filter weight is $E[S(k)\,X(k)]/E[X^2(k)] = 0$, i.e. the output is the primary signal. The 3 dB improvement is consistent with DS beamforming performance against noise uncorrelated between sensors. For finite N, the linear FIR beamformer output always contains some residual noise power while target power is unaffected. Jammer output power goes to zero as $N \to \infty$. Note that the MSE is independent of the probability density function of the jammer as long as the jammer power σ_j^2 is constant. This is generally not true for nonlinear filtering, as will be shown in the following chapters.

The FIR beamformer cannot cancel a sinusoidal jammer at the aliasing frequencies. Let $h(.)$ denote the impulse response of the FIR filter. The filter output is given by

$$Y(k) = \sum_{i=0}^{N} h(i)\,X(k-i), \qquad k \geq 0. \tag{3.30}$$

Because the filter input is zero after the first Δ samples, the output is also zero after the first $\Delta + N$ samples. Consequently, the FIR beamformer transmits the sinusoidal jammer unchanged after a transition phase of $\Delta + N$ samples.

The distance d between microphones influences the noise suppression of the FIR beamformer. If d is small compared to the incident wave lengths, frequencies close to dc cannot be cancelled well. If d is increased sufficiently, noise suppression of low fre-

quency components improves but jammer attenuation becomes small around the spatial aliasing frequencies. The following example will illustrate this effect. A single white jammer with unity variance arrives at the two-microphone GJ beamformer with a delay $\Delta = 1$ and, in a second experiment, with $\Delta = 4$. We assume a sampling frequency of 10 kHz and an angle of arrival $\theta = 45°$. By (3.3), this corresponds to sensor spacings $d = 19.5$ cm and $d = 4.9$ cm for $\Delta = 4$ and $\Delta = 1$, respectively. The parameters d and θ are depicted in Figure 3.2. Figure 3.8 shows power spectra of the beamformer outputs for optimum FIR filters with $N = 4$. Note that the target is set to equal to zero here. For the larger array ($d = 19.5$ cm), the aliasing frequencies are dc, 2500 Hz and 5000 Hz. For the smaller array ($d = 4.9$ cm), the only aliasing frequency is dc. Observe the power peaks around these frequencies. Although the large array cancels frequencies around dc better than the small array, its total output power is only 1/3 as opposed to 1/6 for the small array. As for the ML endfire array, performance *improves* with smaller sensor spacing. This result is consistent with extensive experiments [41].

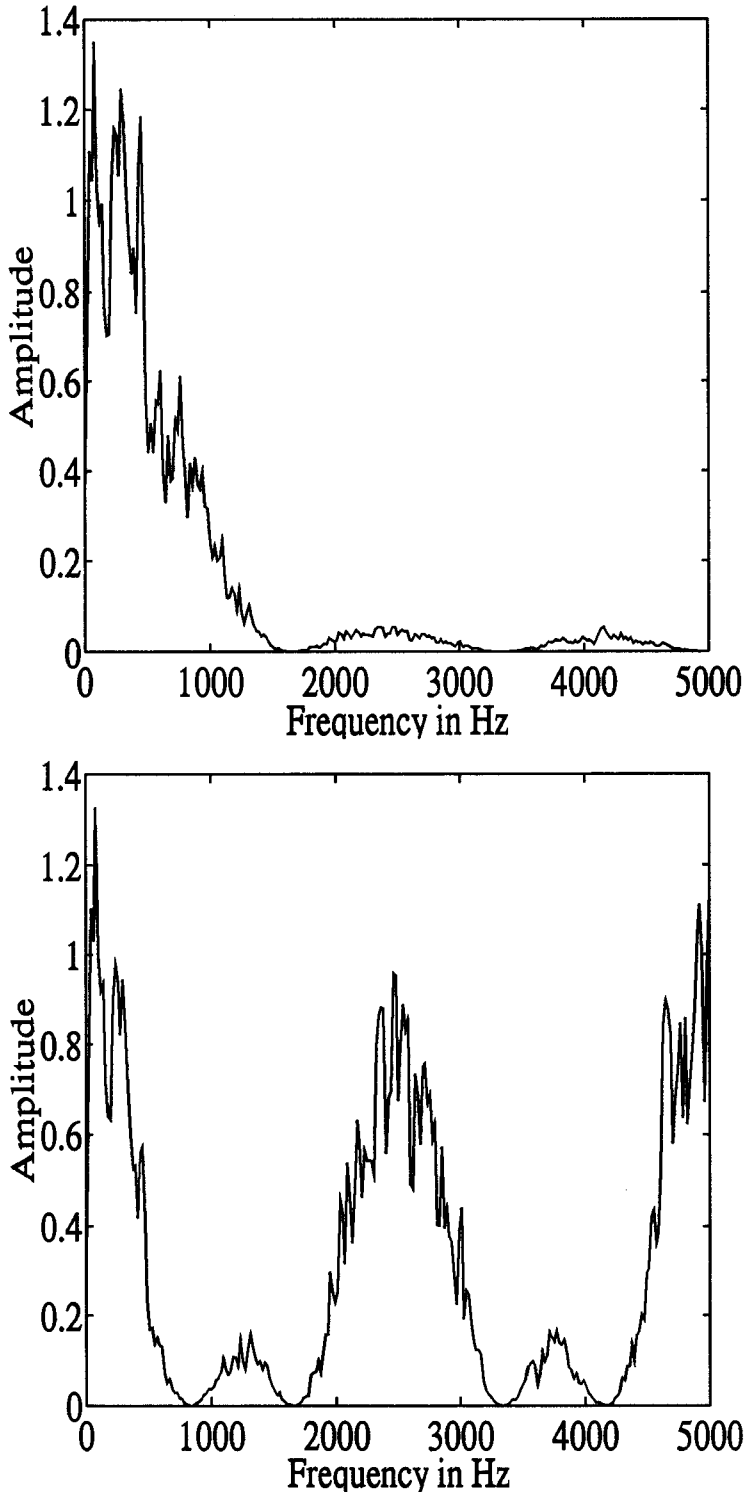
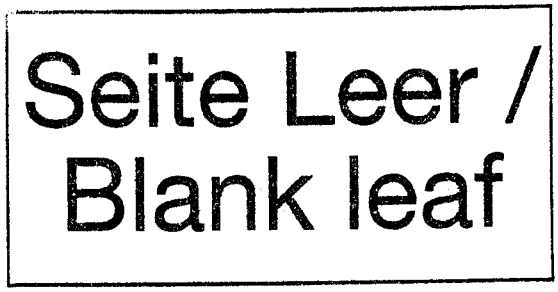


Figure 3.8: Power spectra of GJ beamformer output for sensor distance $d = 4.88$ cm (top) and $d = 19.5$ cm (bottom). The filter length in both cases is $N = 4$.

3.5. Summary of Chapter 3

- To separate the signals from n sound sources, at least n microphones are required.
- The number of grating lobes (number of spatial aliasing frequencies) increases for increasing inter-microphone distance.
- Time-invariant beamforming is well-suited for isotropic noise environments, e.g., highly resonant rooms, parties or restaurants.
- Time-adaptive beamforming is useful for cancelling single directional interferences, e.g., a passing car or single noise sources in weakly resonant rooms.
- Ideally, the two-microphone GJ beamformer with the unconstrained Wiener filter can suppress one directional jammer completely. A more practical realization with a linear FIR filter always results in some residual jammer output power.
- The ML time-invariant and the FIR adaptive beamformers accomplish better jammer suppression for smaller inter-microphone distance. This is particularly useful for hearing-aids where the inter-microphone distance is required to be as small as possible due to cosmetic reasons.


Seite Leer /
Blank leaf

Chapter 4: Nonlinear Adaptive Filters

A-priori knowledge of the conditional probability density is required to calculate the Bayes conditional mean (1.3). Unfortunately, this knowledge is usually not available in real-world situations. Consequently, the Bayes filter function can, at best, be approximated. This chapter describes two prominent nonlinear function approximators; the Volterra filter and the multilayer perceptron. Some other nonlinear structures are mentioned briefly as well, but these were not considered in the experiments.

All nonlinear filters in this chapter can be characterized by the following generic equation:

$$Y(k) = \sum_{i=1}^{M} w_i \, \Psi_i(\boldsymbol{X}(k), \boldsymbol{c}_i), \tag{4.1}$$

where the output $Y(k)$ at discrete time k is a linear combination of the M basis functions $\Psi_i(.,.)$. The arguments of the basis functions $\Psi_i(.,.)$ are the input vector $\boldsymbol{X}(k)$, as defined in (1.1), and a parameter vector $\boldsymbol{c}_i$. The task is to estimate the parameters $w_1, w_2, ..., w_M, \boldsymbol{c}_1, \boldsymbol{c}_2, ..., \boldsymbol{c}_M$ such that the output minimizes some performance criterion. As mentioned in Chapter 1, the minimization criterion used in this thesis is the MSE of (1.2), where $S(.)$ denotes the desired signal and the filter output $Y(.) = \hat{S}(.)$ is an estimate of $S(.)$. Generally, the set of basis functions is chosen *before* filtering and is kept fixed during the filter operation. A judicious choice of a particular set of basis functions may reduce the filter's computational complexity (the number of basis functions) significantly. For example, if the Bayes filter function has the shape depicted in Figure 1.1, the the obvious choice is a sigmoidal basis function, e.g., tanh(.) or arctan(.).

Nonlinear filters will generally produce frequency components which are not present in the input signal. As an example, consider

the simple filter

$$Y(.) = X(.) + X(.)^2. \tag{4.2}$$

For the input $X(k) = A\cos(\omega k)$, this filter generates the output $Y(k) = A\cos(\omega k) + \frac{A^2}{2}\cos(2\omega k) + \frac{A^2}{2}$. System (4.2) has produced energy at the first harmonic of the input as well as a dc offset. If the input contains two sinusoids at frequencies ω_1 and ω_2, namely $X(k) = A\cos(\omega_1 k) + B\cos(\omega_2 k)$, the output becomes

$$\begin{aligned} Y(k) &= X(k) + \frac{A^2}{2}\cos(2\omega_1 k) + \frac{B^2}{2}\cos(2\omega_2 k) + \frac{A^2}{2} + \frac{B^2}{2} \\ &\quad + \frac{AB}{2}\cos([\omega_1 + \omega_2]k) + \frac{AB}{2}\cos([\omega_1 - \omega_2]k). \end{aligned}$$

In addition to the first harmonics and offsets, new *interaction* terms appear at frequencies $\omega_1 + \omega_2$ and $\omega_1 - \omega_2$. When a nonlinear system forms the L^{th} power of a sinusoidal input, the output contains energy at all L harmonics. This property has been used for separating white noise from a square wave signal [65].

Because of the generation of harmonics and interaction terms, nonlinear filtering may introduce aliasing effects. The filter may generate energy at frequencies greater than the Nyquist frequency. These frequency components fold over into the band between zero and the Nyquist frequency. Aliasing can be avoided by choosing the sampling rate for the input signal high enough that the Nyquist frequency is well above the signal's upper band limit. This technique was used in [48] for nonlinearly processed speech. No difference was audible, however, when compared to processing with a Nyquist frequency close to the upper band limit of the input. One possible explanation is that the filter produced relatively little energy beyond the Nyquist frequency so that the resulting frequency aliasing was not audible.

4.1. The Volterra Filter

The polynomial, or Volterra, filter is one of the most popular nonlinear filter realizations. It has been used in various applications including channel equalization, echo and noise cancellation, and distortion analysis in semiconductor devices. For tutorials on this filter and its adaptation, see [66, 67] which also list references for these applications. With respect to model (4.1), the basis functions $\Psi_i(.,.)$ of the polynomial filter are all possible products of the components of the input vector $\boldsymbol{X}(.)$. The Volterra filter of order P comprises all products with up to P factors. What is the total number of basis functions of a P^{th} order filter? Because the input vector contains $N+1$ components, the number of all possible products consisting of p factors is

$$\left(\begin{array}{c} (N+1)+p-1 \\ p \end{array} \right). \tag{4.3}$$

Expression (4.3) represents the number of *combinations with repetitions* of order p which can be constructed from $N+1$ elements. The total number of basis functions M_{Volt} is obtained by summing expression (4.3) over all orders up to P:

$$M_{Volt} = \sum_{p=0}^{P} \left(\begin{array}{c} N+p \\ p \end{array} \right) = \left(\begin{array}{c} N+P+1 \\ P \end{array} \right). \tag{4.4}$$

Equation (4.4) reveals the major disadvantage of the polynomial filter. Even for moderate filter lengths N and low orders P, the number of basis functions (coefficients) is very large. For example, a cubic filter with N=20 has 2024 coefficients!

The P^{th} order Volterra filter output for filter length N is described by

$$Y(k) = w_0 + \sum_{p=1}^{P} \sum_{n_1=0}^{N} \sum_{n_2=n_1}^{N} .. \sum_{n_p=n_{p-1}}^{N} w(n_1,..,n_p) X(k-n_1)..X(k-n_p), \tag{4.5}$$

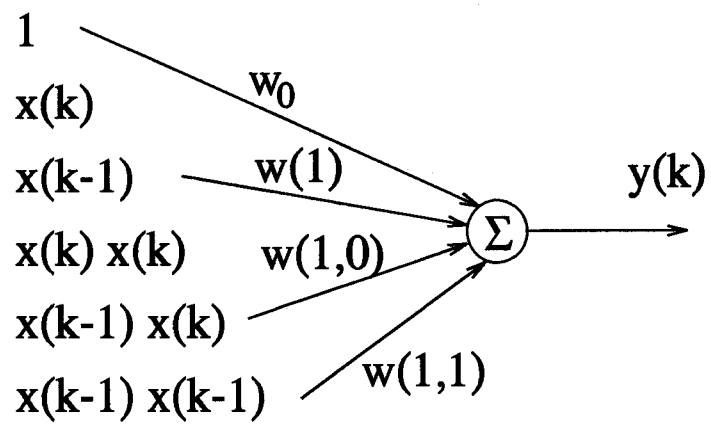


Figure 4.1: Volterra filter example for $N = 1$ and $P = 2$. For simplicity, only some coefficients w(...) are shown.

with $n_0 = 0$. The coefficient w_0 belongs to the basis function $\Psi = 1$ and represents an offset. Equation (4.5) can be written in a more illustrative way:

$$\begin{aligned} Y(k) &= w_0 + \sum_{n_1=0}^{N} w(n_1)\, X(k-n_1) \\ &\quad + \sum_{n_1=0}^{N} \sum_{n_2=n_1}^{N} w(n_1,n_2)\, X(k-n_1)\, X(k-n_2) \\ &\quad \vdots \\ &\quad + \sum_{n_1=0}^{N} \sum_{n_2=n_1}^{N} .. \sum_{n_P=n_{P-1}}^{N} w(n_1,\cdots,n_P)\, X(k-n_1) \cdots X(k-n_P). \end{aligned}$$

For all permutations of a particular index tuple $(n_1, n_2, ...n_P)$, the products $X(k-n_1)X(k-n_2)...X(k-n_P)$ are indistinguishable. To avoid distinct filter coefficients for indistinguishable products, the summation indices n_i in (4.5) start at n_{i-1} for $i = 2...P$.

For sufficiently large P, the filter (4.5) can approximate any continuous function to an arbitrary degree of accuracy [68]. Figure 4.1 shows a second-order example. For $P = 1$, the polynomial filter reduces to the well-known linear FIR filter. In this case, the total number of basis functions (now identity functions) is

$M_{Volt} = N + 2$, and the output becomes

$$Y(k) = w_0 + \sum_{n_1=0}^{N} w(n_1)\, X(k - n_1). \tag{4.6}$$

Volterra basis functions do not contain the argument vector $\boldsymbol{c}_i$. Therefore, the filter output depends only *linearly* on the coefficients $w(...)$. This fact has an important consequence, viz., adaptive Volterra filters can be treated by linear adaptive filter theory. On-line adaptation of the Volterra filter with the standard LMS or RLS algorithm for linear filters is possible.

By comparison to other common nonlinear filters, the MSE surface in coefficient space does not contain any local minima. Consequently, optimum MMSE Volterra filters can be calculated off-line as shown in Section 5.2.

4.2. The Perceptron

The multi-layer perceptron has received increased attention within the engineering community in the last decade. It has been applied to problems in control [69], forecasting [70] and pattern recognition [71]. Further applications can be found in [72]. In pattern classification tasks, the basis functions of the perceptron are usually sigmoids (s-shaped functions) which divide the input space into regions defining the desired class. For the noise filtering and beamforming applications considered in this thesis, it is less obvious that sigmoids are a good choice. However, because a finite set of sigmoidal basis functions can approximate any continuous function arbitrarily well [73], this set has been considered as an alternative to the polynomial set described in the preceding section.

The output of the perceptron in this study is given by

$$Y(k) = \sum_{i=1}^{M_{Perc}} w_i \tanh(\boldsymbol{c}_i^T \boldsymbol{X}(k) - b_i). \tag{4.7}$$

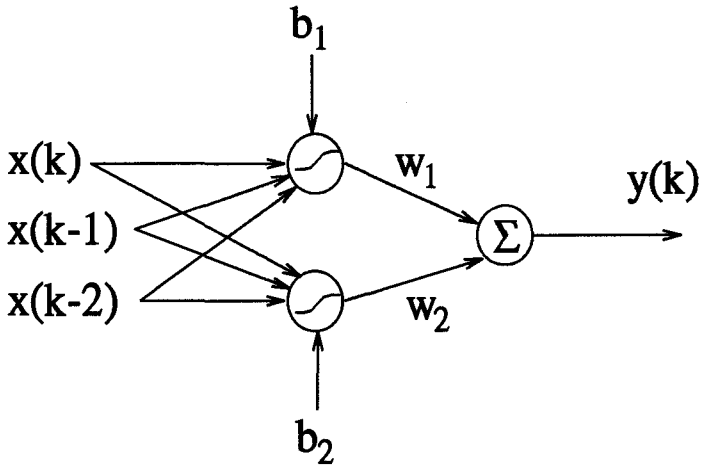


Figure 4.2: Perceptron filter example for $N = 2$ and $M_{Perc} = 2$ so-called hidden units. For simplicity, the coefficient vectors $\vec{c}_i$ are not shown.

As opposed to the Volterra output, the perceptron output *does not* depend linearly on the filter coefficients $\boldsymbol{c}_i$ and b_i. Therefore, the MSE surface in the coefficient space may contain local minima. It is interesting to note that the MSE surface of the perceptron filter (4.7) does not have a unique global minimum due to the odd nonlinearity tanh(.). Suppose that a particular set $\{\boldsymbol{c}_i, b_i, w_i\}_{i=1...M_{Perc}}$ is a solution generating the minimum MSE. If an arbitrary subset i is changed to $\{-\boldsymbol{c}_i, -b_i, -w_i\}$, the output of the network is identical to the output of the network with the original set for all inputs $\boldsymbol{X}(.) \in \Re^{N+1}$. Hence, there are at least $2^{M_{Perc}}$ global minima[1].

The perceptron coefficients can be adapted with the back-propagation algorithm [75], a gradient descent method for non-linear feed forward networks. Note that the global optimization of

[1] After writing these last lines a classical researcher's nightmare occurred to us (fortunately a tiny one). Feeling somewhat proud at having discovered a lower bound on the number of global minima *on our own*, we happened to read article [74]. Not only did Hecht-Nielsen have the same thoughts two years earlier, but he also illuminated another parameter symmetry: Any interchange of two subsets i and j does not change the input-output mapping of the perceptron. This increases the minimum number of global minima to $M_{Perc}!\, 2^{M_{Perc}}$.

the coefficients through backpropagation cannot guarantee finding a global minimum. Two kinds of coefficient updating are common practice. The *batch* method updates the parameters after accumulating the gradients over a set of input vectors. The *on-line* method updates the parameters after each presentation of an input vector. Experiments indicate that both methods exhibit similar convergence speeds for small-scale problems with non-redundant inputs. On-line adaptation is faster for larger training sets containing redundant inputs [76, 77]. In pattern classification, inputs of the same class are redundant when they differ only slightly from each other. A possible application in adaptive beamforming requires on-line processing (or at least mini-batch) to track non-stationary acoustic signals.

To increase the convergence speed of backpropagation, many modifications of the original algorithm have been suggested. Most of these are designed for the batch mode. A tutorial on the most important batch methods and current research on a possible acceleration of on-line convergence is given in [78].

4.3. Other Nonlinear Structures

The architecture of a radial basis function (RBF) network is identical to that of a perceptron, except that the RBF filter employs Gaussian nonlinearities. A brief summary on RBF networks can be found in [71]. The output of this filter at time k is

$$Y(k) = \sum_{i=1}^{M_{RBF}} w_i \exp(-||\boldsymbol{X}(k) - \boldsymbol{c}_i||^2/\sigma_i^2), \qquad (4.8)$$

where the centers $\boldsymbol{c}_i$ determine the locations of the Gaussians in the $(N+1)$-dimensional input space and the parameters σ_i control their widths. For the perceptron, the distribution of the sigmoids in the input space is governed by the bias weights b_i (see equation (4.7)). The main difference between these two commonly used networks is that the radial basis functions are "local" whereas hyperbolic tangents are nonzero almost everywhere

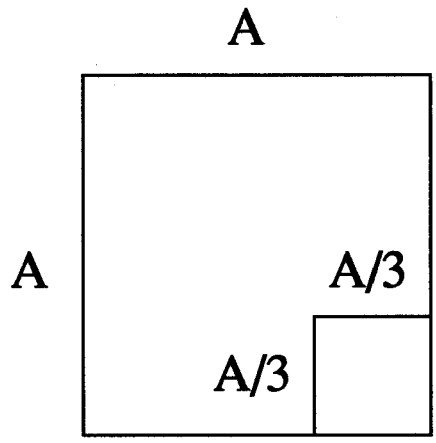


Figure 4.3: The input space is a cube with side length A. The local Gaussian basis function is approximately zero outside the cube with side length A/3.

in the input domain. The following simple example shows that many local basis functions may be needed for covering a high-dimensional input space. Assume that the input vectors uniformly occupy an (N+1)-dimensional cube with side length A. We approximate the region where the radial basis functions are nonzero by an (N+1)-dimensional cube with side length $A/3$. Thus, 3^{N+1} Gaussian basis functions are required to cover the volume A^{N+1} of the input space. Figure 4.3 illustrates the situation for N=1. To reduce the number of basis functions, the widths could be increased but only at the expense of less precision in the RBF approximation.

For the RBF filter, local optimization procedures (subsets of coefficients are adapted separately) have been suggested to decrease the computational load [79]. However, the price for this can be a loss of steady-state performance. Tarassenko and Roberts [80] have verified this statement experimentally.

Another nonlinear filter has been proposed in [81]. The first basis function is the identity function, and the others are given by the piecewise linear mapping $\Psi(\boldsymbol{X}, \boldsymbol{c}_i) = |\boldsymbol{c}_i^T \boldsymbol{X} - 1| - |\boldsymbol{c}_i^T \boldsymbol{X} + 1|$. The arguments $\boldsymbol{c}_i$ are adapted *separately* from the coefficients w_i in (4.1). As mentioned in the context of the RBF filter, such

procedures may lead to a lower approximation accuracy than does a simultaneous optimization of all coefficients.

4.4. Optimum Nonlinear Filters

Any continuous basis function can be approximated by an P^{th}-order polynomial over a finite interval. Replacing all basis functions of a nonlinear filter by polynomials transforms that filter into a Volterra filter. This was illustrated for the perceptron in [48]. It follows that the optimum Volterra filter of sufficiently high order P approximates the upper performance limit of any of the nonlinear filters discussed in this chapter. An experimental verification can be found in [48] and in Section 5.2. However, the Volterra filter has an exorbitant number of coefficients for larger filter lengths ($N > O(10)$) so that an estimation of optimum nonlinear filter performance is only possible for small N.

4.5. Summary of Chapter 4

- All nonlinear filters in this chapter form a linear combination of M nonlinear basis functions.
- If the filter output depends only linearly on the coefficients (e.g. the Volterra filter), linear adaptive filter theory can be used to adapt the nonlinear filter.
- If the output is a nonlinear function of the coefficients (e.g. the perceptron), current adaptation algorithms such as backpropagation are not guaranteed to find the global minimum of the error surface in weight space.
- For small N, the optimum Volterra filter of sufficiently high order P can approximate the upper performance limit of any of the nonlinear filters discussed in this chapter.

Chapter 5: Experiments and Results

Nonlinear filters in adaptive noise cancellers have been considered exclusively for nonlinear distortions of the noise component in the reference signal [82] or in the primary signal [83]. At first glance, it may not be obvious why one benefits from nonlinear filters even in the absence of nonlinear distortions. Section 1.2 explains this fact with the help of Bayes estimation theory. In this chapter, we analyze the two-microphone GJ adaptive beamformer which includes an adaptive noise canceller with a nonlinear filter for processing non-Gaussian signals.

5.1. Optimum Performance

The following analysis refers to the situation depicted in Figure 5.1 (for the reader's convenience, we repeated this figure from Section 3.4), assuming a target-free reference signal (3.14) and a primary signal (3.15). Assuming that target and jammer are statistically independent processes, we find that the Bayes filter (1.3) for the noise canceller is

$$\begin{aligned} E[S(k) \mid \boldsymbol{x}] &= E[T(k) + \frac{1}{2}(J(k) + J(k-\Delta)) \mid \boldsymbol{x}] \\ &= E[T(k)] + E[\frac{1}{2}(J(k) + J(k-\Delta)) \mid \boldsymbol{x}] \\ &= E[\frac{1}{2}(J(k) + J(k-\Delta)) \mid \boldsymbol{x}] \end{aligned} \tag{5.1}$$

where the last line assumes a zero-mean target process. With this assumption, the Bayes filter is independent of the target. The unconstrained Wiener filter is also independent of the target as shown in Section 3.4. Therefore, the target was set to zero without loss of essential generality. The jammer was an i.i.d. process and its inter-microphone delay was taken as $\Delta = 1$.

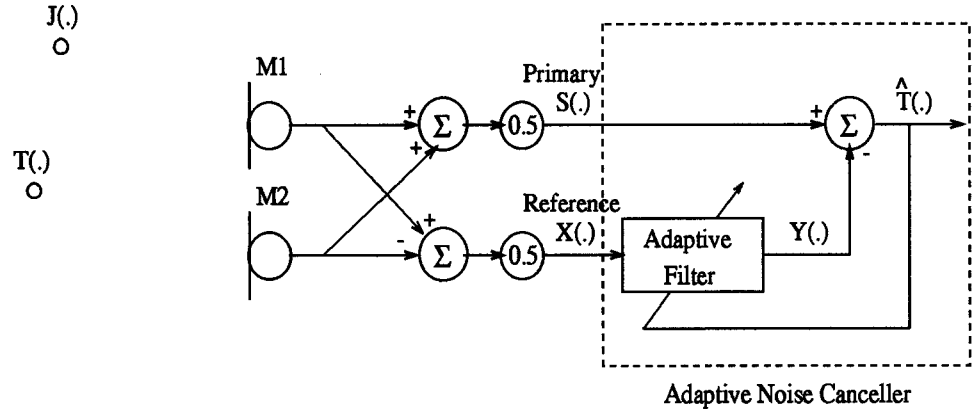


Figure 5.1: Two-microphone Griffiths-Jim beamformer for suppression of directional background noise.

Bayes filters (5.1) and their corresponding MMSEs (1.4) were calculated for the several jammer probability densities listed in Table 5.1. For filter lengths $N > 1$, the Bayes filter function can be a complex expression. But, for the uniformly distributed jammer, a relatively simple expression was obtained in Appendix D :

$$\begin{aligned} \boldsymbol{v}(\boldsymbol{x}) &= 2\,(0\,,\, x(k)\,,\, x(k)+x(k-1)\,,\, \ldots\,,\, \sum_{i=1}^{N+1} x(k-i))^T \\ \hat{s}_B(\boldsymbol{x}) &= \frac{1}{2}(max(\boldsymbol{v})+min(\boldsymbol{v}))-x(k), \end{aligned} \tag{5.2}$$

where $\boldsymbol{v}(.)$ is an $(N+2)$-dimensional auxiliary vector field which must be calculated for every input vector $\boldsymbol{x}$. The operation $max(\boldsymbol{v})$ $(min(\boldsymbol{v}))$ picks the maximum (minimum) component of vector $\boldsymbol{v}$. A similar formula was derived in Appendix D for the one-sided exponentially distributed jammer :

$$\hat{s}_B(\boldsymbol{x}) = max(\boldsymbol{v}) + \frac{1}{N+2} - x(k). \tag{5.3}$$

Note that (5.2) and (5.3) are closed form expressions for arbitrary filter lengths N. The Bayes functions for Laplace-distributed and

Jammer Prob. Density	Definition
One-Sided Exp.	$p_J(j) = \begin{cases} exp(-j) & \text{if } j \geq 0 \\ 0 & \text{else} \end{cases}$
Gamma	$p_J(j) = \begin{cases} j\,exp(-j) & \text{if } j \geq 0 \\ 0 & \text{else} \end{cases}$
Uniform	$p_J(j) = \begin{cases} 1/2\,a & \text{if } j \epsilon [-a, a] \\ 0 & \text{else} \end{cases}$
Laplace	$p_J(j) = \frac{1}{2} exp(-\lvert j \rvert)$
Gaussian	$p_J(j) = (2\pi\sigma_j^2)^{-1/2}\, exp(-(j - m_j)^2/2\sigma_j^2)$

Table 5.1: Definitions of jammer probability densities.

Gamma-distributed jammers turned out to be very long expressions and, therefore, are not included here. The optimum filter for the Gaussian jammer is linear and can be obtained by solving the Wiener-Hopf equations. To illustrate the nonlinear shapes of the Bayes functions, Figure 5.2 shows two examples for a two-dimensional input vector. This figure indicates that different jammers may require different basis functions to approximate (to a desired degree) the Bayes function with a minimum number of coefficients. In the case of the uniform jammer, a set of piecewise linear basis functions is apparently well-suited to this task. For the Laplacian jammer, smoother nonlinearities may be required. It is pointed out that the Bayes function for the Laplacian jammer is "almost linear". This is consistent with the fact that the Bayes filter does not perform significantly better than the Wiener filter as can be seen in Figure 5.3.

Table 5.2 summarizes the Bayes MMSEs resulting from the $(N + 2)$-dimensional integrations (1.4). The derivation of these formulas can be found in Appendix E. Note that the normalized MSEs in this table represent the *residual* jammer output powers of the beamformer. The normalization by $\sigma_s^2 = \sigma_j^2/2$ re-

stricts the MSE to the interval between zero and one, where a normalized MSE of one indicates a filter output of zero. The jammer output power of the optimum linear filter assuming a zero-mean jammer is derived in Appendix C. Linear filtering performs identically for arbitrary jammer distributions because it employs identical second-order correlations. Since the exponentially distributed jammer has a non-zero mean, the linear filter must include a *bias weight* to reach MSE$= 2/(N+2)$. If the bias weight is not included, the normalized MSE will increase by an offset equal to $E^2[J]/\sigma_s^2$.

Figure 5.3 depicts the results of Table 5.2 together with the performance curves for Laplace-distributed and Gamma--distributed jammers. These two curves were obtained by implementing the Bayes filter, filtering a signal of 100,000 samples and then averaging the squared beamformer output samples. We could not carry out the integration (1.4) for these distributions because of the complexity of the corresponding Bayes filter functions.

To obtain a first acoustic impression, we implemented the Bayes filter (5.2) and the optimum linear filter for the uniformly distributed jammer. The target signal, spoken by a male person and sampled at 8 kHz, was the sentence "This is your two-microphone hearing aid". The broadband input TJR was -20.3 dB at each of the microphones. For $N = 40$, optimum linear and nonlinear processing achieved output TJRs of -4.3 dB and 4.2 dB, respectively. Figure 5.4 displays spectrograms of the original target signal and of the optimum linear and nonlinear beamformer outputs. Observe the increased reduction of residual noise components in Figure 5.4 (bottom). Since frequencies near zero appear almost in phase at the two microphones, both beamformers had difficulty suppressing low frequency interference, although cancellation was better in the nonlinear case. The residual noise of the Bayes beamformer sounded like "crackling" while that of the linear beamformer sounded like lowpass-filtered white noise. The nonlinear TJR improvement (8.5 dB) was clearly audible.

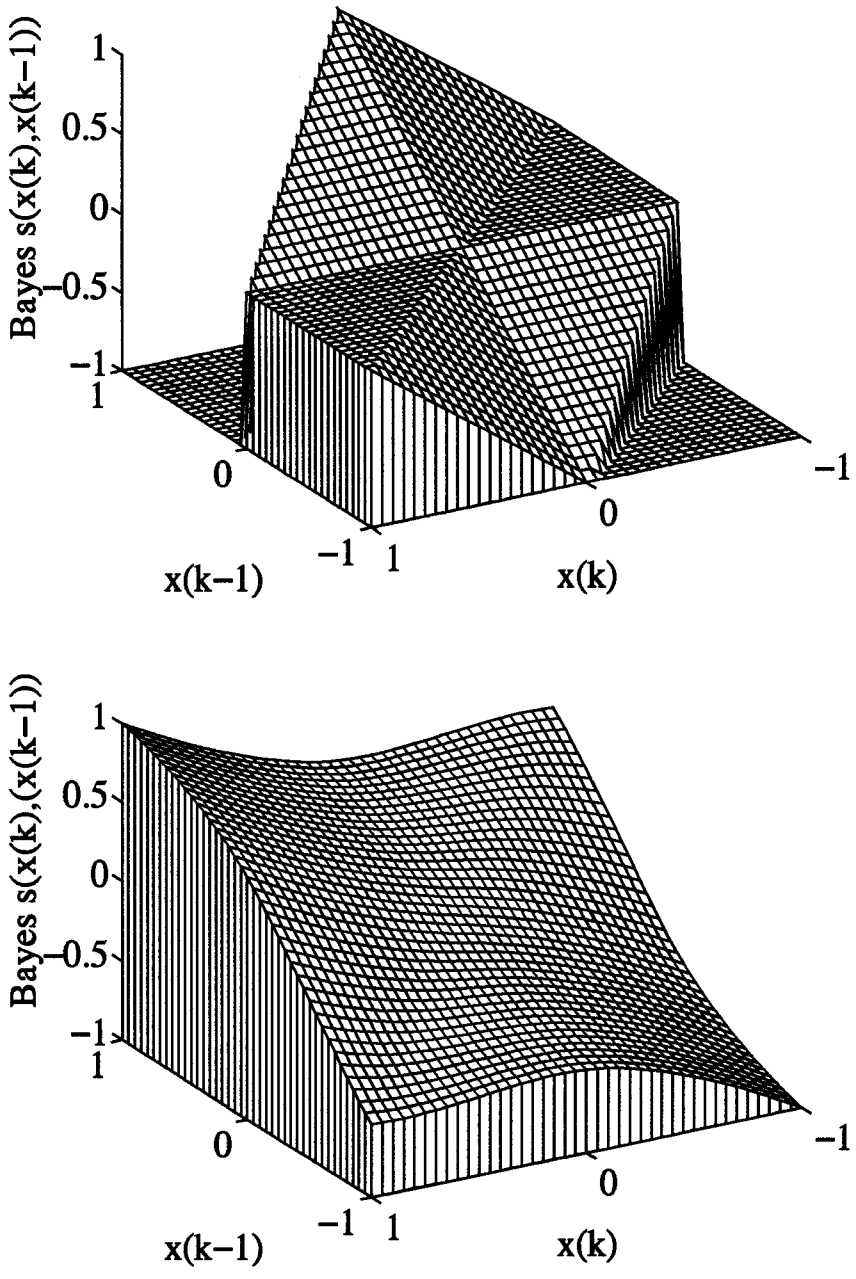


Figure 5.2: Bayes filter function for a uniformly distributed jammer (top) and a Laplacian distributed jammer (bottom). The filter length is $N = 1$.

Jammer Prob. Density	Linear MSE	Bayes $MMSE$
One-Sided Exp.	$\frac{2}{N+2}$	$\frac{2}{(N+2)^2}$
Uniform	$\frac{2}{N+2}$	$\frac{12}{(N+3)(N+4)}$
Gaussian	$\frac{2}{N+2}$	$\frac{2}{N+2}$

Table 5.2: MSEs of optimum linear and nonlinear beamformers for i.i.d. jammers with three different probability densities as a function of filter length N. All expressions are normalized by the variance σ_s^2 of the primary signal.

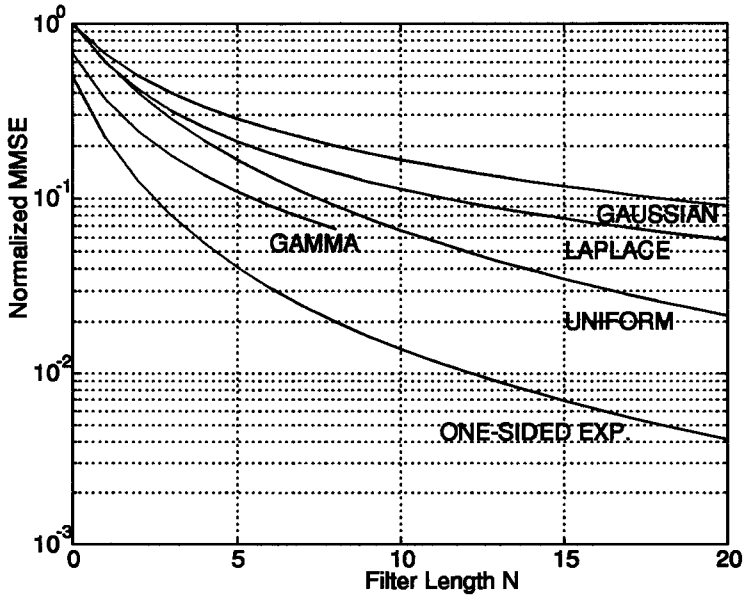


Figure 5.3: Normalized Bayes MMSEs for various jammer probability densities versus filter length N.

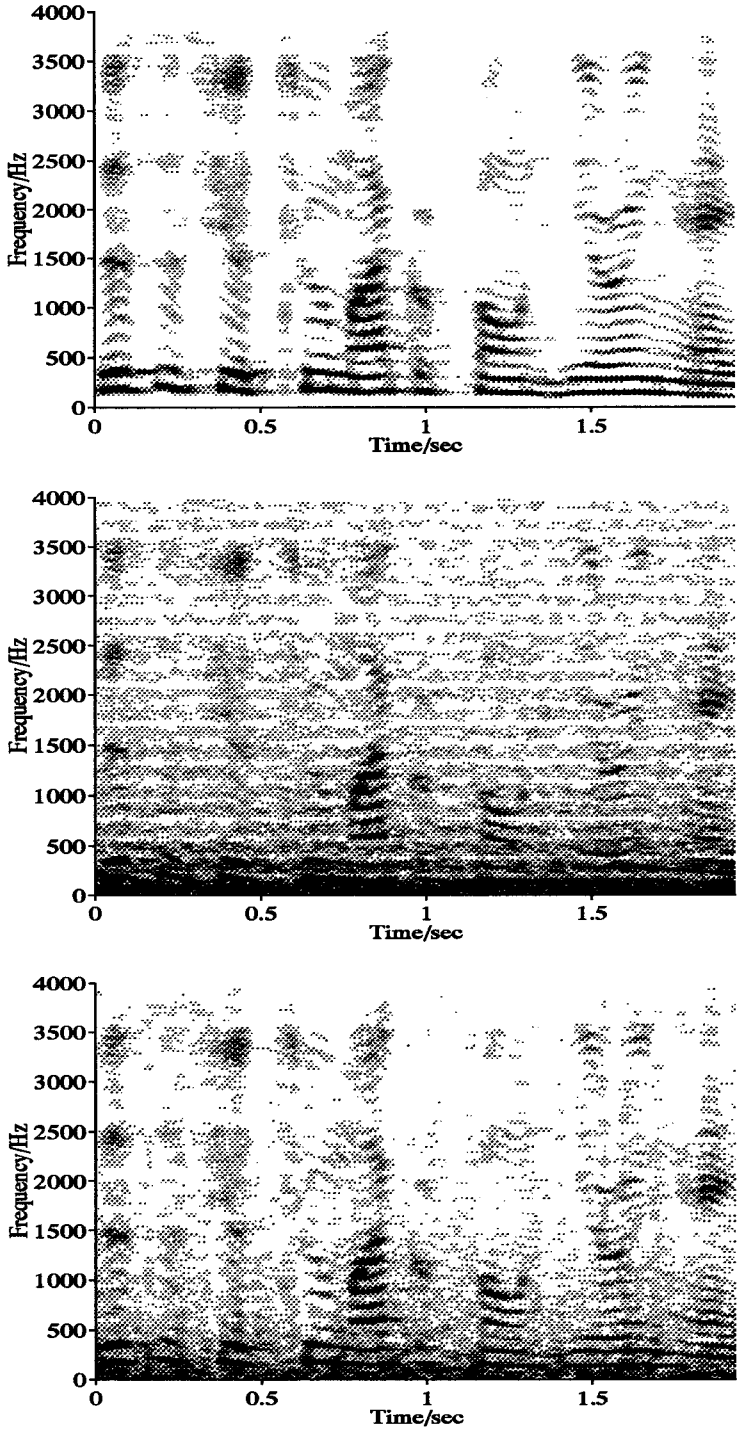


Figure 5.4: Spectrogram of target sentence (top). Spectrogram of optimum linear beamformer output for $N = 40$ (middle). Spectrogram of optimum nonlinear beamformer for $N = 40$ (bottom).

5.2. An Off-line Experiment with I.I.D. Noise

This section describes the performance of the Volterra and perceptron filters in an off-line experiment. The goal was to approximate the Bayes performance curve in Figure 5.3 for the uniform i.i.d. jammer. The target signal was set to zero.

To find the optimum Volterra coefficients for a given order P, we first rewrite equation (4.5) as

$$Y(k) = \boldsymbol{w}_e^T \, \boldsymbol{X}_e, \tag{5.4}$$

where

$$\begin{aligned} \boldsymbol{w}_e^T &= (w_0, w(0), ..., w(N), w(0,0), ..., w(N, ..., N)) \\ \boldsymbol{X}_e &= (1, X(k), ..., X(k-N), X^2(k), ..., X^P(k-N))^T. \end{aligned} \tag{5.5}$$

The subscript e stands for "extended". Similar to linear filter theory, the optimum coefficients solve the "extended" Wiener-Hopf equations

$$E[\boldsymbol{X}_e \, \boldsymbol{X}_e^T] \, \boldsymbol{w}_e = E[\boldsymbol{X}_e \, S]. \tag{5.6}$$

The input correlation matrix on the left side of (5.6) was estimated by

$$E[\boldsymbol{X}_e \, \boldsymbol{X}_e^T] \approx \frac{1}{L} \sum_{k=1}^{L} \boldsymbol{x}_e(k) \, \boldsymbol{x}_e^T(k). \tag{5.7}$$

The cross correlation vector on the right side of (5.6) was estimated by

$$E[\boldsymbol{X}_e \, S] \approx \frac{1}{L} \sum_{k=1}^{L} \boldsymbol{x}_e(k) \, s(k). \tag{5.8}$$

Steiner and Joho showed in [84] that, for i.i.d. jammers with symmetric probability density functions, the Volterra coefficients belonging to *even* order components of $\boldsymbol{X}_e$ vanish. Hence, a third-order Volterra filter was employed but without second-order terms. Using a 10,000-point uniformly distributed jammer sequence ($L = 10,000$), the optimum coefficients were obtained

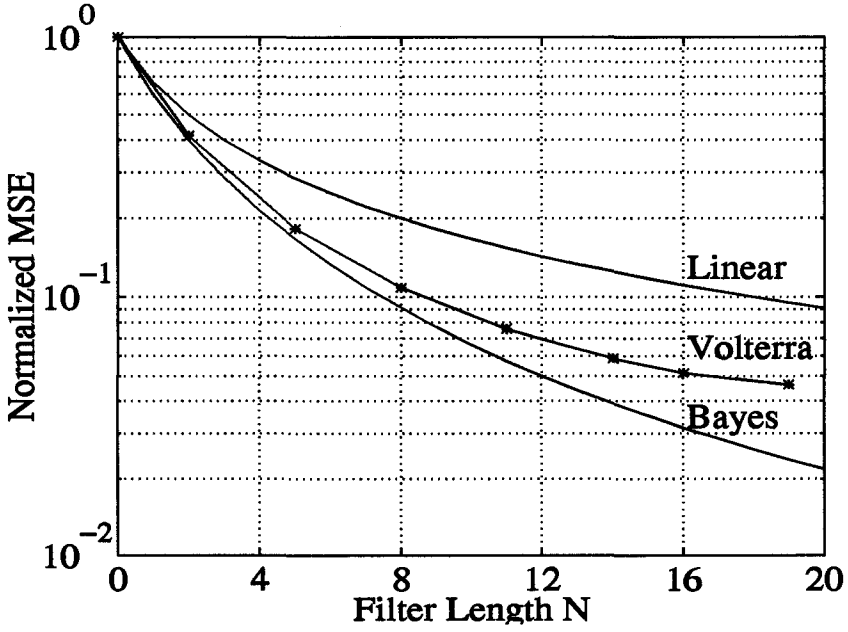


Figure 5.5: Normalized jammer power at the beamformer output for the optimum third-order Volterra filter versus filter length N.

by solving (5.6) with the time averages (5.7) and (5.8). A new test sequence of 100,000 jammer samples (again uniformly distributed) was processed by the beamformer with the fixed optimum weights. The MSE was determined by averaging the squared beamformer output samples of the test run and is depicted in Figure 5.5 together with optimum linear and nonlinear MSEs. Results for other jammer distributions can be found in [46]. Computing the optimum weights required 3.5 hours for $N = 16$ on a SUN SPARCstation 10. It should be noted that beyond $N = 16$, the number of coefficients became so large that the computation of optimum coefficients exhausted the memory of the SUN workstation (The measurement at $N = 19$ was performed on a HP workstation 735 with more memory).

Currently, no methods exist to determine the optimum coefficients of the perceptron. Consequently, it was not possible to draw the optimum performance curve. We conjecture that an extensive backpropagation training results in nearly optimum perceptron performance. It must be emphasized, however, that per-

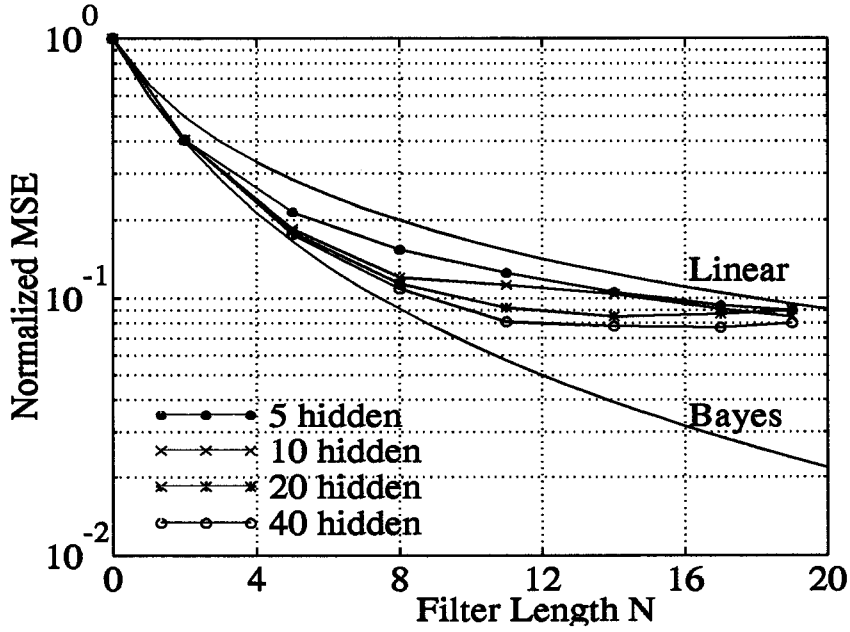


Figure 5.6: Normalized jammer power at the beamformer output for perceptron filters with 5, 10, 20 and 40 hidden units versus filter length N.

formance depends on the initial coefficients, the learning rate and the amount of training.

The perceptron coefficients were initialized randomly in the interval $[-1, 1]$ and adapted by backpropagation (on-line mode) without a momentum term. During training, a jammer signal of 500,000 samples was presented to the beamformer 100 times. The learning rate was fixed at 0.001. Four different filters with 5, 10, 20 and 40 hidden units were examined. After training, a new test sequence of 100,000 jammer samples was processed with frozen weights. The results are summarized in Figure 5.6. The data in this figure were averaged over two independent runs with two different sets of initial weights. Perceptron performance degraded strikingly for filter lengths between 10 and 20. Extensive additional training improved the situation. For example, presenting the training signal 300 times to the filter with $N = 19$ and 40 hidden units resulted in a normalized MSE = 0.056, representing a 30% decrease of MSE. However, the computational burden of

this method increased substantially for larger networks. Training the system with $N = 19$ and 40 hidden units (300 x 500,000 iterations) required about 82 hours on our SUN workstation. It is possible that other training methods may produce better results for $N > 10$.

To summarize, off-line computed coefficients of the perceptron and the optimum Volterra filter could be obtained only for small filter lengths ($N < 20$). Either too little computer memory (Volterra) or too long computation times (perceptron) prohibited simulations beyond $N > 20$. The performance of the Volterra filter was always equal to or better than the performance of the perceptron filters with 5, 10, 20 and 40 hidden units. This was consistent with the concept in Section 4.4.

5.3. An On-line Experiment with I.I.D. Noise

The adaptive filter in a beamforming hearing aid must converge sufficiently fast to adapt to the changing environment and to compensate for head movements. The experiments in this section compare the convergence speed and steady state performance of the on-line adapted perceptron and Volterra filter. Although the test involved only one particular jammer (uniform i.i.d. noise) at $N = 8$, the results reflect a typical filter behavior that was observed also in other simulations with different filter lengths and signals.

The jammer delay between the microphones was set again to $\Delta = 1$. The target signal was female speech (one sentence) sampled at 8 kHz and the input TJR was zero dB. The perceptron was adapted with on-line backpropagation and the third-order (without second-order terms) Volterra coefficients were adjusted with the standard LMS and RLS algorithms. For the exact formulas, see Haykin [85] on page 332 (LMS) and on page 485, Table 13.2 (RLS). Because the jammer was stationary, the RLS forgetting factor λ was set to unity.

When the target is present, the learning rate of the backprop-

Filter	Maximum Learning Rate	Steady State MSE	Off-Line MSE
Linear FIR, LMS	0.02	0.2068	0.2000
Percep. $M_{Perc} = 5$	0.01	0.1866	0.1550
Percep. $M_{Perc} = 20$	0.005	0.1962	0.1140
Volterra, LMS	0.01	-	0.1085
Linear FIR, RLS		0.2050	0.2000
Volterra, RLS		0.1131	0.1085

Table 5.3: Steady state and off-line normalized MSEs of various linear and nonlinear filters. For the perceptron with hidden units, the entries are "quasi" steady state MSEs (see text). The off-line results were taken from the previous section. The filter length was $N = 8$.

agation (or LMS) algorithm must stay below a certain threshold to avoid target cancellation. This threshold is generally not identical with the maximum learning rate which would render the adaptive filter unstable. For linear filters, this form of target cancellation is discussed in more detail in [50] starting on page 429.

The maximum learning rates for the perceptron without hidden units[1], with 5 and 20 hidden units and for the LMS-adapted Volterra filter were determined as follows. The beamformer was run with a series of different learning rates. For each learning rate, we listened to the *filter output* (not the beamformer output) and chose the maximum rate for which the target signal was not audible in the output. The maximum learning rates are shown in Table 5.3. Using these learning rates ensured an undistorted target signal at the beamformer output. Larger learning rates would have allowed a faster adaptation at the expense of additional target distortion.

[1]The perceptron without hidden units is defined to be a linear FIR filter. In this case, backpropagation reduces to the LMS algorithm.

Because the beamforming did not affect the target, it was set to zero in the subsequent experiments. The beamformer was run ten times employing ten different sets of initial coefficients and ten different uniformly-distributed jammer signals for each filter in Table 5.3. For the perceptron, the coefficients were initialized with the simple and effective method described in [86]. The linear FIR and the Volterra coefficients were chosen from a normal distribution with mean zero and variance one-quarter. Figure 5.7 depicts the ensemble-averaged learning curves as a function of the sample index. The steady state normalized MSEs in Table 5.3 were estimated from these curves by *time-averaging* the instantaneous squared errors from sample index 10,000 to index 30,000. For the RLS algorithm, the averages were calculated between the indices 1,000 and 20,000.

The perceptron (5 and 20 hidden units) learning curves in Figure 5.7 appear to reach the steady state after about 10,000 iterations. In a test simulation of 60,000 iterations, however, the MSE decreased further. For example, between the indices 40,000 and 60,000, the MSE of the perceptron with 20 hidden units declined to 0.1843. Because the perceptron error decayed very slowly over many iterations, the entries in Table 5.3 are called "quasi" steady state MSEs. In other simulations, the perceptron error varied slowly over several 100,000 iterations without reaching a final value [87]. With a sampling rate of 8 kHz, the perceptron required more than one second to reach a quasi stationary state. In a hearing aid, slow changes of quasi steady state performance after "convergence" would be of minor importance. It is striking that the perceptron did not perform significantly better than the linear FIR filter. Giulieri obtained similar results in [87]. He carried out experiments with speech jammers and various filter lengths N and could not find significant improvements over the linear filter.

The LMS-adapted Volterra filter converged extremely slowly. The MSE (measured in blocks of 20,000 samples) still decreased after 100,000 iterations. The RLS Volterra filter converged af-

ter approximately 1,000 iterations with a steady state MSE close to its optimum value, but the computational burden of this algorithm is the highest of all algorithms in this section. For the linear filter, the RLS algorithm requires $O(N^2)$ operations per iteration [88]. The third-order Volterra filter has $O(N^3)$ coefficients and thus requires $O(N^6)$ operations per iteration. Backpropagation with M_{perc} hidden units has a complexity of $O(M_{Perc}N)$ operations per iteration. Despite the high computational burden of the Volterra filter, the results suggested that it was worthwhile to examine this filter further as we do in the next section.

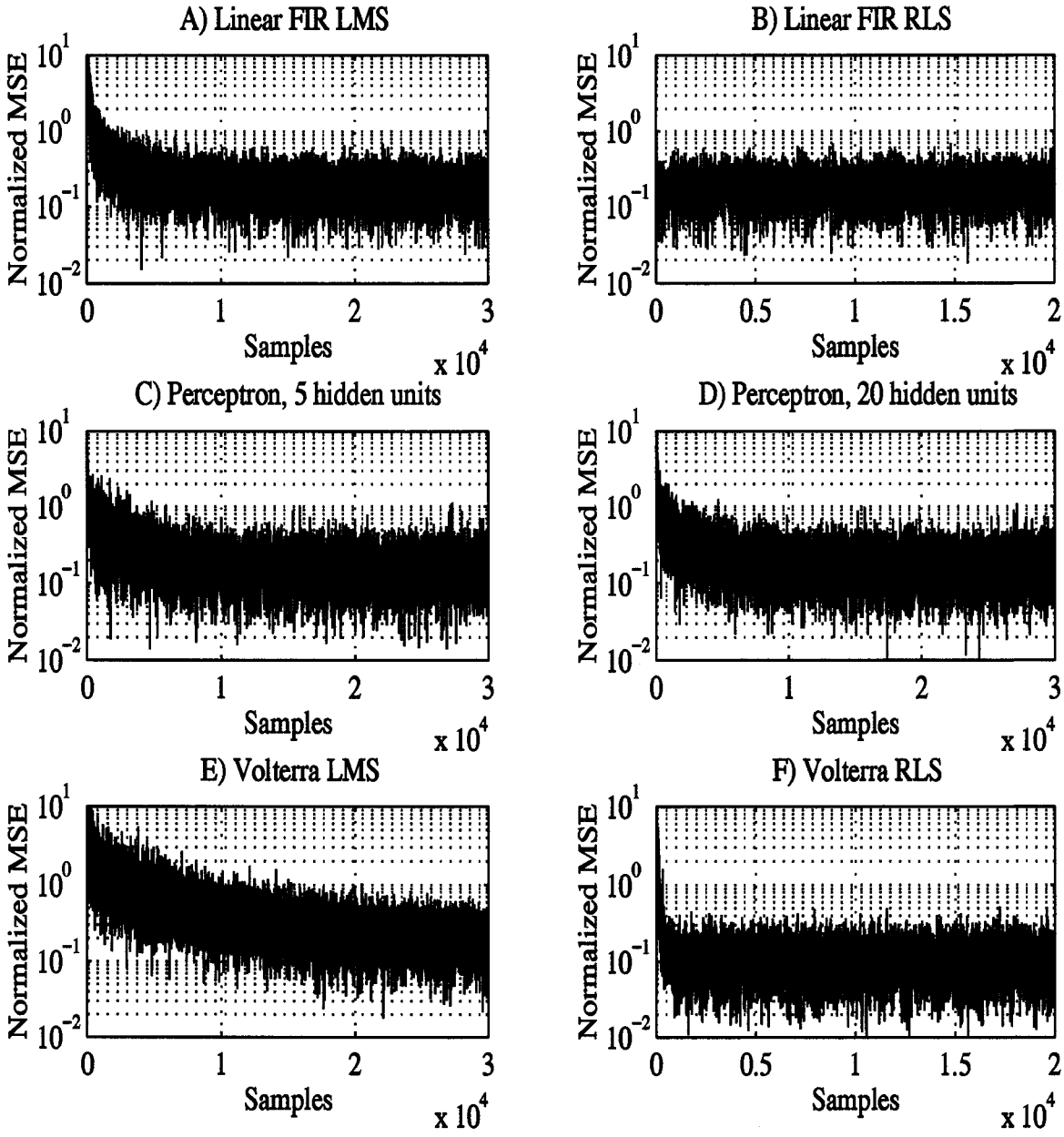


Figure 5.7: Ensemble averaged learning curves for various on-line adaptive linear and nonlinear filters. Note the different abscissa scaling for the RLS-adapted filters.

5.4. Volterra Beamforming with a Speech Jammer

This section examines the Volterra beamformer processing speech signals. The experimental set-up was as follows. The target and jammer signals were male speech, sampled at 8 kHz and pre-emphasized with the high-pass filter given in [42]. The jammer delay between the microphones was set to $\Delta = 4$. The broadband input TJR was -10 dB (measured at a single microphone) and normally-distributed white noise was added to each sensor at a jammer-to-sensor noise ratio of 40 dB. The GJ beamformer was run with the second-order Volterra filter and the linear FIR filter. In preliminary tests, the RLS-adapted nonlinear filter exhibited instabilities at the transients between voiced and unvoiced sounds and performed worse than the linear filter. For this reason, the optimum coefficient vector $\boldsymbol{w}_e(k)$ at each time step k was calculated by solving the normal equations (compare to equation (5.6))

$$\boldsymbol{R}_e(k)\, \boldsymbol{w}_e(k) = \boldsymbol{P}_e(k) \tag{5.9}$$

directly by Gaussian elimination [89]. Here, $\boldsymbol{R}(k)$ denotes the input auto correlation matrix , calculated recursively by

$$\boldsymbol{R}_e(k) = \lambda\, \boldsymbol{R}_e(k-1) + \boldsymbol{x}_e(k)\, \boldsymbol{x}_e^T(k), \tag{5.10}$$

and $\boldsymbol{P}(k)$ designates the cross correlation vector

$$\boldsymbol{P}_e(k) = \lambda\, \boldsymbol{P}_e(k-1) + s(k)\, \boldsymbol{x}_e(k), \tag{5.11}$$

where λ is the forgetting factor. Solving (5.9) directly increased the number of operations per iteration from $O(N^4)$ to $O(N^6)$ for the Volterra filter.

In the preceding experiments, performance was measured with the MSE. For a prediction of speech intelligibility, a more appropriate metric is required that takes account of perceptual aspects of hearing. The intelligibility-weighted gain G_I is a convenient measure to estimate the beamformer's impact on speech intelligibility [22, 90]. It is defined by

$$G_I = \sum_i w(i)\, [\mathrm{TJR}_{out}(i) - \mathrm{TJR}_{in}(i)], \tag{5.12}$$

where $\mathrm{TJR}_{out}(i)$ and $\mathrm{TJR}_{in}(i)$ are the target-to-jammer ratios of the i^{th} third-octave frequency band in dB at the beamformer output and input, respectively. The weights $w(i)$ reflect the contribution of each band to intelligibility. For the calculation of G_I in this work, the weights were taken from [43]. Peterson [22] showed that G_I estimates the change of the speech reception threshold[2] (SRT) through the beamformer for normal-hearing test persons and continuous speech. A positive G_I indicates an intelligibility improvement, whereas a negative gain represents an intelligibility loss.

Unfortunately, the measure (5.12) can be computed only for the nonlinear beamformer when the target signal remains unchanged by the processing. When the target is affected by the beamforming (e.g. a target misalignment will result in target components in the reference channel causing target cancellation), $\mathrm{TJR}_{out}(i)$ cannot be determined for the nonlinear beamformer. Even in the case of a target-free reference channel, target cancellation may occur for high learning rates as discussed in Section 5.3. A similar behavior was observed for the optimum on-line filters in this section when the forgetting factor λ was too small. The smaller λ, the less data were used in the time-averaged cross correlation between the primary signal and the filter input vector. If the time average was based on too little data, the target in the primary became correlated with the filter input, causing target cancellation.

The filters were adapted on-line with zero initial coefficients while target and jammer were present. The coefficients at each time step were copied into an identical beamformer processing only the jammer. The output of this "slaved" beamformer was used for the calculation of G_I. The beamformer was run with various forgetting factors λ and filter lengths N. At each run, we listened to the output of the slaved beamformer to ensure

[2]The SRT is the target-to-jammer ratio at which 50% of the target speech is intelligible. For example, a gain $G_I = 10$ dB means that the SRT for the unprocessed signals must be 10 dB higher than for the processed signals.

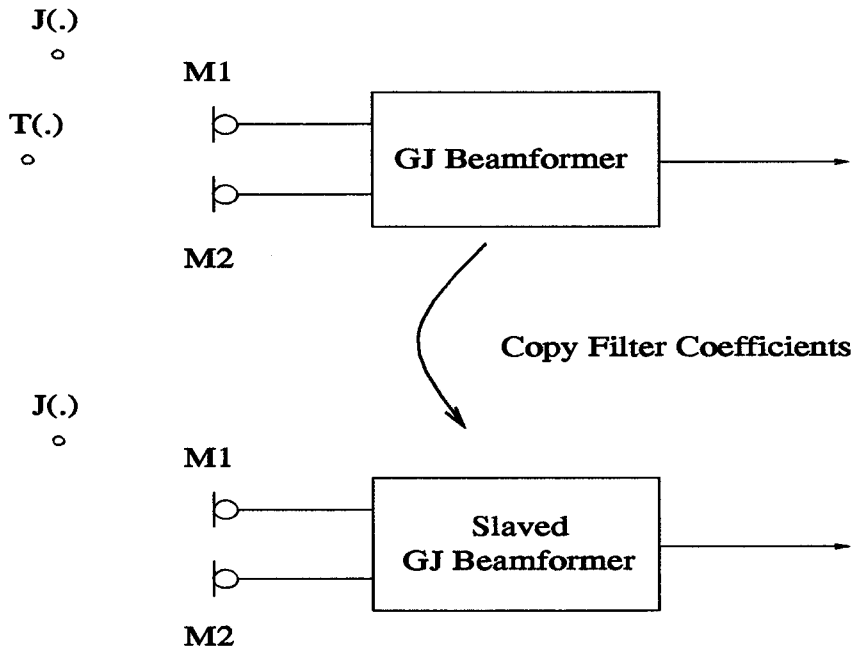


Figure 5.8: Illustration of the procedure for monitoring target cancellation and calculating G_I.

that no target components were audible. Figure 5.8 illustrates this procedure. Let $T_{in}(i)$ ($T_{out}(i)$) denote the input (output) target power level of dB in the i^{th} third-octave frequency band. Assuming that the procedure in Figure 5.8 prevents target cancellation, then $T_{in}(i)$ is approximately equal to $T_{out}(i)$ for all i, and the gain in equation (5.12) can be approximated as follows

$$\begin{aligned} G_I &= \sum_i w(i)\,[T_{out}(i) - J_{out}(i) - T_{in}(i) + J_{in}(i)] \\ G_I &\approx \sum_i w(i)\,[J_{in}(i) - J_{out}(i)], \end{aligned} \tag{5.13}$$

where $J_{in}(i)$ and $J_{out}(i)$ are the jammer power levels in dB of the i^{th} third-octave band of the input and output, respectively. The input jammer power was determined from the primary channel such that performance was measured against that of the fixed beamformer with uniform weights. All jammer power spectra were computed from the signal samples between the indices 10,000 and 28,000. The power spectra were calculated with the MATLAB^{TM} function SPECTRUM.

λ	N=5	N=10	N=20
1	4.4 (3.6)	7.1 (7.0)	11.3 (12.4)
0.999	4.3 (3.3)	8.3 (7.1)	12.8 (13.1)
0.998	4.4 (3.2)	8.8 (7.3)	-
0.997	4.6 (3.2)	9.1 (7.5)	-
0.996	4.7 (3.3)	9.3 (7.6)	-
0.995	4.9 (3.3)	9.4 (7.7)	-
0.990	5.4 (3.5)	-	-

Table 5.4: Intelligibility-weighted gains G_I in dB for the second-order Volterra beamformer for various filter lengths N and forgetting factors λ. The numbers in parentheses are the gains obtained with the linear FIR beamformer.

The results in Table 5.4 were calculated according to (5.13). This table contains empty entries in those cases where audible target cancellation occurred. Interestingly, linear and nonlinear filters distorted the target for the same parameter sets (N, λ). For filter lengths of 5 and 10 , the nonlinear filter performed only marginally better than the linear FIR filter. Note that the benefits of nonlinear processing increased for declining forgetting factors. This is consistent with the discussion in the introduction of this thesis. The filter with a finite memory ($\lambda < 1$) could exploit the non-Gaussian short-term statistics of speech. For larger N, the forgetting factor had to be increased to avoid audible target cancellation. A large number of second-order terms in the input vector requires more data to cancel out the target in the cross correlation vector.

For $N = 20$, the linear filter outperformed the Volterra filter. Although the target was very weak at the input (TJR= -10 dB), its impact on the estimation of the Volterra coefficients increased for larger N. To see to what extent the target in the primary channel influences the gain G_I, the experiment was repeated without

λ	N=5	N=10	N=20
1	4.5 (3.6)	7.3 (7.0)	12.4 (12.5)
0.999	4.4 (3.3)	8.6 (7.1)	14.6 (13.2)
0.998	4.5 (3.2)	9.2 (7.4)	15.7 (13.2)
0.997	4.8 (3.3)	9.9 (7.6)	16.9 (13.3)
0.996	5.0 (3.3)	10.4 (7.8)	17.9 (13.4)
0.995	5.2 (3.4)	10.8 (8.0)	18.7 (13.5)
0.990	5.9 (3.7)	12.4 (8.5)	22.1 (13.7)

Table 5.5: Intelligibility-weighted gains G_I in dB for the second-order Volterra beamformer for various filter lengths N and forgetting factors λ and no target in the primary channel. The numbers in parentheses are the gains obtained with the linear FIR beamformer.

the target and summarized in Table 5.5. These figures represent the maximum performance gains of the second-order Volterra filter over the linear filter given the (hypothetical) assumption that the target does not influence the calculation of the coefficients. A comparison of Tables 5.4 and 5.5 shows that the linear filter was less sensitive to the target than the Volterra filter, especially for smaller forgetting factors. But particularly for smaller forgetting factors, nonlinear processing exhibited better jammer cancellation than the linear filter. In other words, the advantage of the nonlinear filter was diminished by increased target sensitivity. At $N = 20$, this effect more than neutralized the benefit of nonlinear processing as can be seen in Table 5.4.

The processed speech was also judged by informal listening. The speech jammer of the nonlinear beamformer seemed to be more distorted and slightly reduced in power compared to that of the linear processor. Because the improvements of G_I were modest, a formal listening test was not performed.

Finally, experiments have been performed with the same pa-

rameters N and λ, but with higher input TJRs and other jammer delays Δ. Because of its high target sensitivity, the Volterra filter advantage decreased for larger input TJRs. For example, at input TJR = 0 dB, the nonlinear improvements did not exceed 0.5 dB. Decreasing the delay Δ gave similar results. For $\Delta = 1$, the nonlinear processor could not outperform the linear filter. One possible explanation is suggested by Figure 5.4. The jammer energy near the spatial aliasing frequency at zero Hz could be cancelled more efficiently by the nonlinear filter. If nonlinear processing is most beneficial around spatial aliasing frequencies, the reduction of Δ from four to one (i.e. the spatial aliasing frequency at the normalized frequency $\pi/2$ was removed) would diminish the nonlinear advantage. This effect is amplified by the intelligibility weighting because the weights are greater at higher frequencies.

Chapter 6: Summary and Discussion

Currently, array processing is one of the most promising techniques for reducing background noise in hearing aids. While most research in this area has been conducted using linear digital filters, we concerned ourselves in this thesis with nonlinear filters. When the adaptive filter in the two-microphone adaptive Griffiths-Jim beamformer is adjusted to minimize its mean squared error, nonlinear filtering is warranted for non-Gaussian signals. Our most important findings are:

- At fixed filter length N, optimum Bayes and adaptive nonlinear filters achieved smaller MSEs than the corresponding linear filters for non-Gaussian jammer signals.

- For a single i.i.d. jammer, the improvement (measured in decrease of MSE) of the Bayes filter over the Wiener filter increased for larger filter lengths.

- For filter lengths $N < 10$, the perceptron could approximate the optimum Bayes filter. For larger N, the off-line training method used in this thesis was too computation-intensive. On-line processing with the backpropagation algorithm was not sufficiently fast in the adaptive beamforming application.

- Optimum third-order Volterra filters could be computed from a training signal for filter lengths $N < 20$. Computer memory requirements prohibited the calculations beyond $N = 20$. The Volterra filter could be adjusted to operate fast enough with the RLS algorithm and attained a satisfactory steady state MSE. The computational load of the third-order Volterra RLS, however, was prohibitive for $N > 16$.

- The Volterra filter outperformed the perceptron in all experiments.

- Through fading memory ($\lambda < 1$), the second-order Volterra beamformer could exploit the non-Gaussian short-term statistics of a speech jammer. Table 5.5 shows that the improvements of the nonlinear system in jammer cancellation became larger for increasing N. Simultaneously, target cancellation intensified as a trade-off.

- The second-order Volterra beamformer exhibited a higher target sensitivity than the linear system. Assuming that no audible target cancellation occured, nonlinear processing improved the intelligibility-weighted gain by maximally 2 dB relative to linear filtering for a speech jammer. The computational complexity of the second-order Volterra filter did not justify this performance gain.

It remains an open question whether computationally efficient adaptive nonlinear filters exist which can adapt sufficiently fast and provide significant performance gains over linear filters. The simulations suggest that the improvement through nonlinear processing increases for larger filter lengths. But the realization of a "practical" nonlinear filter becomes more difficult for greater N. In situations with simple stationary signals, it may be possible to calculate optimum nonlinear filters, as was demonstrated for the uniformly distributed jammer. Such filters may be implemented in a computationally efficient way, justifying their performance gain relative to the linear filter.

At small filter lengths ($N < 20$), one may find other basis functions exceeding the performance of the second-order Volterra filter in the speech jammer experiment. For example, a pilot test with a third-order Volterra filter revealed higher gains at fixed N, but also higher target sensitivity than with the second-order filter.

Currently, only linear adaptive filters can be realized for larger N. They certainly "do the job", but it is left to future work to

investigate whether realizable long nonlinear filters exist which might do better. In some cases, long filters may not be desired. For example, long filters in the Frost beamformer [50] will render the system more sensitive to target cancellation in a resonant environment. Here, short filters are required to prevent target reflections (which are considered as jammer) from being correlated with the direct target. The same problem exists in the GJ beamformer with a conventional non-zero primary delay (not shown in Figure 5.1).

Future work should concentrate on finding sets of *problem-dependent* basis functions to minimize computational complexity. One suggestion can be found in [91] for unimodal error surfaces. Simultaneously, the convergence speed of the Volterra LMS could be accelerated by decorrelating its input. Adaptive filters with nonlinear error surfaces like the perceptron are more difficult to analyze. Convergence speed may be improved by using second-order methods (e.g. the Levenberg-Marquardt algorithm) in a sliding mini-batch.

Appendix A: Abbreviations and Symbols

c	Velocity of Sound
d	Spacing between Sensors
Δ	Jammer Delay between Sensors
DS	Delay and Sum
FIR	Finite Impulse Response
GJ	Griffiths-Jim
$\boldsymbol{H}$	Transfer Function Matrix. $H_{ij}(f)$ is the transfer function from source j to sensor i.
i.i.d.	Independent Identically Distributed
J(.)	Jammer Stochastic Process
λ	Wave Length
LMS	Least Mean Square
ML	Maximum Likelihood
M_{Perc}	Number of Perceptron Basis Functions
MSE	Mean Squared Error
M_{Volt}	Number of Volterra Basis Functions
m	Number of Microphones in Array
n	Number of Sound Sources
N	Filter Length
RLS	Recursive Least Squares
SNR	Signal-to-Noise Ratio
T(.)	Target Stochastic Process
TJR	Target-to-Jammer Ratio

Seite Leer /
Blank leaf

Appendix B: Bayes Filter Example

In this appendix, we give the calculations for the Bayes filter example described in the Introduction 1.2. Using equation (1.3), the Bayes filter is

$$\begin{aligned}
\hat{s}_B(x) &= \int_{-\infty}^{+\infty} s\, p_{S|X}(s|x)\, ds \\
&= \int_{-\infty}^{+\infty} s\, \frac{p_{S,X}(s,x)}{p_X(x)}\, ds \\
&= \int_{-\infty}^{+\infty} s\, \frac{p_{S,N}(s,x-s)}{p_X(x)}\, ds \\
&= \frac{\int_{-\infty}^{+\infty} s\, p_{S,N}(s,x-s)\, ds}{\int_{-\infty}^{+\infty} p_{S,N}(s,x-s)\, ds}.
\end{aligned} \tag{B.1}$$

Since $S(.)$ and $N(.)$ are independent processes, the joint probability density function in (B.1) can be expressed as the product of the individual probability densities $p_S(s)$ and $p_N(n)$. Finally, the integration yields

$$\hat{s}_B(x) = \frac{1}{2}\, \frac{e^{x+\frac{1}{4}}\,(erf(x+\frac{1}{2})-1) + e^{-x+\frac{1}{4}}\,(erf(x-\frac{1}{2})+1)}{e^{x+\frac{1}{4}}\,(-erf(x+\frac{1}{2})+1) + e^{-x+\frac{1}{4}}\,(erf(x-\frac{1}{2})+1)}, \tag{B.2}$$

where $erf(.)$ is the error function defined as

$$erf(x) = 2/\sqrt{\pi} \int_0^x exp(-t^2)\, dt.$$

Appendix C: MSE For Linear FIR Filter

In this appendix, we compute the output power of the optimum linear FIR beamformer with N taps. The jammer is white noise and the delay between is set to $\Delta = 1$.

The $(N+1) \times (N+1)$ auto-correlation matrix is

$$\boldsymbol{R} = E[\boldsymbol{X}(k)\boldsymbol{X}^T(k)] = \frac{\sigma_j^2}{4} \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix} = \frac{\sigma_j^2}{4}\boldsymbol{D} \tag{C.1}$$

and the cross-correlation vector is

$$\boldsymbol{P} = E[S(k)\boldsymbol{X}(k)] = \frac{\sigma_j^2}{4}(0 \;\; 1 \;\; 0 \;\; \cdots \;\; 0)^T, \tag{C.2}$$

where σ_j^2 denotes the variance of the jammer. The optimum linear FIR filter achieves

$$MSE = E[S^2] - \boldsymbol{P}^T\boldsymbol{R}^{-1}\boldsymbol{P} = \sigma_t^2 + \frac{\sigma_j^2}{2} - \frac{\sigma_j^2}{4} D_{22}^{-1}. \tag{C.3}$$

The variance of the target is denoted by σ_t^2. Since only the second component of the cross-correlation vector is non-zero, the calculation of $\boldsymbol{P}^T\boldsymbol{R}^{-1}\boldsymbol{P}$ requires only the (2,2) element of the inverse of $\boldsymbol{D}$, denoted by D_{22}^{-1}. This element is

$$D_{22}^{-1} = \frac{(-1)^{2+2}\, det(\boldsymbol{D}'_{22})}{det(\boldsymbol{D})}. \tag{C.4}$$

The $N \times N$ matrix $\boldsymbol{D}'_{22}$ is obtained from $\boldsymbol{D}$ by deleting the second column and the second row. Using Laplace's theorem for determinants and subscripts for the matrix dimension, one finds the

recursion

$$det(\boldsymbol{D}_{N+1}) = 2\, det(\boldsymbol{D}_N) - det(\boldsymbol{D}_{N-1}) \tag{C.5}$$

for $N \geq 2$. For the initial conditions $det(\boldsymbol{D}_1) = 2$ and $det(\boldsymbol{D}_2) = 3$, it follows that

$$det(\boldsymbol{D}_{N+1}) = N + 2. \tag{C.6}$$

Using Laplace's theorem again for the determinant in the numerator of (C.4) gives

$$det(\boldsymbol{D}'_{22}) = 2\, det(\boldsymbol{D}_{N-1}) = 2\, N. \tag{C.7}$$

Inserting (C.6) and (C.7) into (C.3) leads to the formula for the MSE as a function of the filter length N:

$$MSE = \sigma_t^2 + \sigma_j^2 \frac{1}{N+2}. \tag{C.8}$$

Appendix D: Bayes Filters for the GJ Beamformer

This appendix derives optimum nonlinear filters in the two-microphone GJ beamformer for one directional i.i.d. jammer process. Two jammer probability density functions are considered: the uniform and the one-sided exponential density function. These calculations were originally provided by Steiner and Joho in [84] and independently by Kütükçüoglu in [92] for the uniformly distributed jammer. The target signal is assumed to be zero.

The task is to compute

$$E[S|\boldsymbol{X}=\boldsymbol{x}] = \int_{-\infty}^{+\infty} s\, p_{S|\mathbf{X}}(s|\boldsymbol{x})\, ds = \int_{-\infty}^{+\infty} s\, \frac{p_{S,\mathbf{X}}(s,\boldsymbol{x})}{p_{\mathbf{X}}(\boldsymbol{x})}\, ds. \quad \text{(D.1)}$$

The joint probability density function $p_{S,\mathbf{X}}(.\,,.)$ will now be manipulated in several steps.

First, we exploit the fact that the jammer is an i.i.d. random process. The variables $(S, \boldsymbol{X})$ can be expressed as

$$\begin{pmatrix} S \\ \boldsymbol{X} \end{pmatrix} = \underbrace{\frac{1}{2}\begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -1 \end{pmatrix}}_{=\boldsymbol{A}} \underbrace{\begin{pmatrix} J(k) \\ J(k-1) \\ \vdots \\ J(k-N) \\ J(k-N-1) \end{pmatrix}}_{=\boldsymbol{J}}. \quad \text{(D.2)}$$

Because the random vector $(S\ \boldsymbol{X}^T)^T$ is a linear transformation of the random vector $\boldsymbol{J}$, their probability density functions are

related as follows [16] :

$$p_{S,\mathbf{X}}(s,\boldsymbol{x}) = \frac{1}{|det(\boldsymbol{A})|}\, p_{\mathbf{J}}(\boldsymbol{A}^{-1}\begin{pmatrix} s \\ \boldsymbol{x} \end{pmatrix}). \tag{D.3}$$

After calculating $det(\boldsymbol{A}) = (-2)^{-N-1}$ and $\boldsymbol{A}^{-1}$, equation (D.3) becomes

$$\begin{aligned} p_{S,\mathbf{X}}(s,\boldsymbol{x}) \;=\; & 2^{N+1}\, p(s+x(k))\, p(s-x(k)) \\ & p(s-x(k)-2x(k-1)) \cdots \\ & \cdots\, p(s-x(k)-2\,x(k-1)-\cdots-2\,x(k-N)). \end{aligned} \tag{D.4}$$

In the last equation, we used the fact that the jammer joint probability density function $p_{\boldsymbol{J}}(.)$ is the product of $N+2$ identical jammer density functions $p(.)$ (recall that the jammer is an i.i.d. random process). To simplify notation, the subscript J is omitted here.

The second step is a change of variables given by

$$\begin{pmatrix} Z \\ \boldsymbol{X} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & & 0 & 1 \end{pmatrix}}_{=\boldsymbol{B}} \begin{pmatrix} S \\ \boldsymbol{X} \end{pmatrix}. \tag{D.5}$$

Using this transformation and $det(\boldsymbol{B}) = 1$, equation (D.4) becomes

$$\begin{aligned} p_{Z,\mathbf{X}}(z,\boldsymbol{x}) \;=\; & 2^{N+1}\, p(z)\, p(z-2\,x(k)) \\ & p(z-2\,x(k)-2x(k-1)) \cdots \\ & \cdots\, p(z-2\,x(k)-2\,x(k-1)-\cdots-2\,x(k-N)). \end{aligned} \tag{D.6}$$

Note that the integration in (D.1) will now be performed over z instead over s. By the substitution rule for integrals, the integral limits remain and, because $s = z - x(k)$, it follows $dz = ds$.

The third step is another transformation of variables

$$\begin{pmatrix} Z \\ V(k) \\ \vdots \\ V(k-N) \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 2 & \cdots & 2 \end{pmatrix}}_{=\boldsymbol{C}} \begin{pmatrix} Z \\ X(k) \\ \vdots \\ X(k-N) \end{pmatrix}. \quad \text{(D.7)}$$

With $det(\boldsymbol{C}) = 2^{N+1}$, the joint density function (D.6) simplifies to

$$p_{Z,\mathbf{V}}(z,\boldsymbol{v}) = p(z)\, p(z-v(k))\, p(z-v(k-1)) \cdots p(z-v(k-N)). \quad \text{(D.8)}$$

Changing variables $\boldsymbol{X} \rightarrow \boldsymbol{V}$, where $\boldsymbol{V} = (V(k), \cdots, V(k-N))^T$, does not effect the integration in (D.1). With (D.8), the Bayes filter (D.1) becomes

$$E[Z|\boldsymbol{V}=\boldsymbol{v}] = \frac{\int_{-\infty}^{+\infty} z\, p_{Z,\mathbf{V}}(z,\boldsymbol{v})\, dz}{\int_{-\infty}^{+\infty} p_{Z,\mathbf{V}}(z,\boldsymbol{v})\, dz}. \quad \text{(D.9)}$$

The joint probability density function in (D.8) is the product of identical functions $p(.)$ shifted by the values $v(k), \cdots, v(k-N)$. If the jammer probability density function $p(.)$ is not continuous, the integrands in (D.9) will not be continuous. Therefore, the region of integration must be divided into subintervals where the integrands are continuous.

Uniform Jammer

Let $p(.)$ be the uniform density function which is constant in the interval $[-a, a]$ and zero outside this interval. Then, the integrands in (D.9) are nonzero in a specific interval determined as follows. We find the minimum (maximum) component of the

vector $(0 \ \boldsymbol{v}^T)^T$, denoted by v_{min} (v_{max}). Then, the Bayes filter (D.9) is given by

$$E[Z|\boldsymbol{V}=\boldsymbol{v}] = \frac{\int_{v_{max}-a}^{v_{min}+a} z\,(1/2\,a)^{N+2}\,dz}{\int_{v_{max}-a}^{v_{min}+a} (1/2\,a)^{N+2}\,dz} = \frac{1}{2}(v_{min}+v_{max}). \tag{D.10}$$

If S is to be estimated instead of Z, the Bayes filter becomes

$$\hat{s}_B(\boldsymbol{x}) = \frac{1}{2}(v_{min}+v_{max}) - x(k), \tag{D.11}$$

where the values v_{min} and v_{max} can expressed through the components of $\boldsymbol{X}$.

One-sided Exponential Jammer

For the one-sided exponentially distributed jammer, (D.9) is

$$\begin{aligned} E[Z|\boldsymbol{V}=\boldsymbol{v}] &= \frac{\int_{v_{max}}^{+\infty} z\ exp[-z(N+2)+\sum_{i=0}^{N} v(k-i)]\,dz}{\int_{v_{max}}^{+\infty} exp[-z(N+2)+\sum_{i=0}^{N} v(k-i)]\,dz} \\ &= v_{max} + \frac{1}{N+2}. \end{aligned} \tag{D.12}$$

Similarly to (D.11), the Bayes filter can be written as

$$\hat{s}_B(\boldsymbol{x}) = v_{max} + \frac{1}{N+2} - x(k). \tag{D.13}$$

Appendix E: MSEs for Bayes Filters

This appendix calculates the MMSEs of the Bayes filters for two jammer probability densities: the uniform and the one-sided exponential density. It is a summary of the original derivation by Steiner and Joho [84]. The target is set to zero.

In order to compute

$$MMSE = \int_{-\infty}^{+\infty} ds \int_{R^{N+1}} d\boldsymbol{x}\, (s - \hat{s}_B(\boldsymbol{x}))^2\, p_{S,\mathbf{X}}(s, \boldsymbol{x}), \qquad \text{(E.1)}$$

we perform a change of integration variables

$$\begin{pmatrix} z \\ \boldsymbol{v} \end{pmatrix} = \boldsymbol{C}\,\boldsymbol{B} \begin{pmatrix} s \\ \boldsymbol{x} \end{pmatrix}, \qquad \text{(E.2)}$$

where the matrices $\boldsymbol{C}$ and $\boldsymbol{B}$ are defined in Appendix D in equations (D.7) and (D.5), respectively. This results in

$$MMSE = \int_{-\infty}^{+\infty} dz \int_{R^{N+1}} d\boldsymbol{v}\, (z - \hat{z}_B(\boldsymbol{v}))^2\, p_{Z,\mathbf{V}}(z, \boldsymbol{v}), \qquad \text{(E.3)}$$

where $\hat{z}_B(\boldsymbol{v}) = E[Z|\boldsymbol{V} = \boldsymbol{v}]$ and $p_{Z,\mathbf{V}}(z, \boldsymbol{v})$ is defined in Appendix D, equation (D.6) .

For a non-continuous joint density $p_{Z,\mathbf{V}}(z, \boldsymbol{v})$, the integration over z can only be performed over regions where $p_{Z,\mathbf{V}}(z, \boldsymbol{v})$ is continuous. These regions depend on the shifts given by the components of vector $\boldsymbol{v}$.

One-sided Exponential Jammer

The function $p_{Z,\mathbf{V}}(z, \boldsymbol{v})$ is only non-zero for $z > v_{max}$, where v_{max} is the maximum component of the vector $(0\ \boldsymbol{v}^T)^T$. Each of the $(N + 2)$ components of this vector can be the maximum. Hence,

the R^{N+1} can be divided into $(N+2)$ *non-overlapping* subregions where z is integrated from v_{max} to infinity and each component of $\boldsymbol{v}$ is integrated from -infinity to v_{max}. It can be shown that the integration over each of the $(N+2)$ subregions yields the same value. Therefore, the total MMSE is $(N+2)$ times the value of the integration over an arbitrary subregion. A convenient choice is the subregion defined by $v_{max} = 0$. According to (D.12), we have

$$E[Z|\boldsymbol{V} = \boldsymbol{v}] = \frac{1}{N+2}. \tag{E.4}$$

For the exponential jammer, the probability density is

$$p_{Z,\mathbf{V}}(z,\boldsymbol{v}) = \begin{cases} 0 & \text{if } z < 0 \\ exp(-(N+2)z + \sum_{i=0}^{N} v(k-i)) & \text{if } z \geq 0. \end{cases} \tag{E.5}$$

Hence, the total MMSE in (E.3) becomes

$$MMSE = (N+2) \int_{-\underline{\infty}}^{\underline{0}} e^{\sum_{i=0}^{N} v(k-i)}\, d\boldsymbol{v} \int_{0}^{\infty} (z - \frac{1}{N+2})^2\, e^{-(N+2)z}\, dz, \tag{E.6}$$

where the symbols $\underline{\infty}$ and $\underline{0}$ mean that each component of vector $\boldsymbol{v}$ is integrated from -infinity to zero. Carrying out this integration yields

$$MMSE = (N+2)\, \frac{1}{(N+2)^3}. \tag{E.7}$$

Finally, the MMSE is normalized by the variance of the primary signal $\sigma_s = 1/2$ resulting in

$$MMSE = \frac{2}{(N+2)^2}. \tag{E.8}$$

Uniform Jammer

For the uniform jammer, the probability density $p_{Z,\mathbf{V}}(z, \boldsymbol{v})$ is defined by

$$p_{Z,\mathbf{V}}(z, \boldsymbol{v}) = \begin{cases} (\frac{1}{2\,a})^{N+2} & \text{if } (v_{max} - a) < z < (v_{min} + a) \\ & \quad \text{and } (v_{max} - v_{min}) < 2\,a \\ 0 & \text{else.} \end{cases} \tag{E.9}$$

There are $(N+2)(N+1)$ possibilities to pick a maximum and a minimum from the $(N+2)$ elements of vector $(0\ \boldsymbol{v}^T)^T$. As in the previous case with the exponential jammer, the integrations over each of these $(N+2)(N+1)$ subregions yield the same value. Hence, an arbitrary subregion can be chosen for the integration; here we take $v_{min} = 0$ and $v_{max} = v(k)$. From (D.10), it follows for the Bayes estimator

$$E[Z|\boldsymbol{V} = \boldsymbol{v}] = \frac{1}{2}v(k). \tag{E.10}$$

Then, the total MMSE becomes

$$MMSE = (N+2)(N+1) \int_0^{2\,a} dv(k) \int_{\underline{0}}^{\underline{v(k)}} dv(k-2) \cdots dv(k-N) \int_{v(k)-a}^{a} (z - \frac{v(k)}{2})^2 \left(\frac{1}{2\,a}\right)^{N+2} dz, \tag{E.11}$$

where the symbols $\underline{0}$ and $\underline{v(k)}$ are defined similarly as in the case with the exponential jammer. Carrying out this integration gives

$$MMSE = \frac{2\,a^2}{(N+3)(N+4)}. \tag{E.12}$$

Again, the result is normalized by the variance of the primary signal $\sigma_s = a^2/6$, yielding

$$MMSE = \frac{12}{(N+3)(N+4)}. \tag{E.13}$$

Bibliography

[1] T.C. Smedley and R.L. Schow. Frustrations with Hearing Aid Use: Candid Reports from the Elderly. *The Hearing Journal*, 43(6):21–27, June 1990.

[2] A. Parving and B. Philip. Use and Benefit of Hearing Aids in the Tenth Decade and Beyond. *Audiology*, 30:61–69, 1991.

[3] W.E. Leary. Hearing-Aid Makers Are Warned on False Claims. The New York Times, April 26, 1993.

[4] S.F. Boll. Suppression of Acoustic Noise in Speech Using Spectral Subtraction. *IEEE Trans. Acoustics, Speech and Signal Processing*, 27(2):113–120, April 1979.

[5] T.W. Parsons. Separation of Speech from Interfering Speech by Means of Harmonic Selection. *Journal of the Acoustical Society of America*, 60(4):911–918, October 1976.

[6] J.S. Lim. *Speech Enhancement.* Prentice-Hall, 1983.

[7] Y. Ephraim. Statistical-Model-Based Speech Enhancement Systems. *Proceedings of the IEEE*, 80(10):1526–1555, October 1992.

[8] J.M. Kates. Speech Enhancement Based on a Sinusoidal Model. *Journal of Speech and Hearing Research*, 37:449–464, April 1994.

[9] W. Etter. *Contributions to Noise Suppression in Monophonic Speech Signals.* PhD thesis, Institut für Signal und Informationsverarbeitung, ETHZ, Zürich, Switzerland, 1993.

[10] W. Soede, F.A. Bilsen, and A.J. Berkhout. Development of a Directional Hearing Instrument Based on Array Technology. *Journal of the Acoustical Society of America*, 94(2):785–798, August 1993.

[11] W. Soede, F.A. Bilsen, and A.J. Berkhout. Assessment of a Directional Microphone Array for Hearing-Impaired Listeners. *Journal of the Acoustical Society of America*, 94(2):799–808, August 1993.

[12] M.W. Hoffman, T.D. Trine, K.M. Buckley, and D.J. Van Tasell. Robust Adaptive Microphone Array Processing for Hearing Aids: Realistic Speech Enhancement. *Journal of the Acoustical Society of America*, 96(2):759–770, August 1994.

[13] J.E. Greenberg. *Improved Design of Microphone-Array Hearing Aids.* PhD thesis, Research Lab of Electronics, Massachusetts Institute of Technology, Cambridge, USA, September 1994.

[14] H. Bamberger. Phonak: Mit Hörgeräten an die Weltspitze. *Technische Rundschau*, 46, 1994.

[15] J.L. Hennessy and N.P. Jouppi. Computer Technology and Architecture: An Evolving Interaction. *IEEE Computer Magazine*, 24(9):18–30, September 1991.

[16] J.L. Massey. Mathematische Grundlagen der Nachrichtentechnik. Vorlesungsskript, Institut für Signal und Informationsverarbeitung, ETHZ, Zürich, Switzerland, October 1992.

[17] W.B. Davenport. An Experimental Study of Speech-Wave Probability Distributions. *Journal of the Acoustical Society of America*, 24(4):390–399, July 1952.

[18] L.R. Rabiner and R.W. Schafer. *Digital Processing of Speech Signals.* Prentice-Hall Signal Processing Series. Prentice-Hall, 1978.

[19] I.N. Bronstein and K.A. Semendjajew. *Taschenbuch der Mathematik.* Verlag Harri Deutsch, 1985.

[20] R.J. Webster. Ambient Noise Statistics. *IEEE Transactions on Signal Processing*, 41(6):2249–2253, June 1993.

[21] L.J. Griffiths and C.W. Jim. An Alternative Approach to Linearly Constrained Adaptive Beamforming. *IEEE Trans. Antennas and Propagation*, 30(1):27–34, 1982.

[22] P.M. Peterson. *Adaptive Array Processing for Multiple Microphone Hearing Aids.* PhD thesis, Research Lab of Electronics, Massachusetts Institute of Technology, Cambridge, USA, February 1989.

[23] J.S. Lim, A.V. Oppenheim, and L.D. Braida. Evaluation of an Adaptive Comb Filtering Method for Enhancing Speech Degraded by White Noise Addition. *IEEE Trans. Acoustics, Speech and Signal Processing*, 26(4):354–358, August 1978.

[24] R.J. Stubbs and Q. Summerfield. Algorithms for Separating the Speech of Interfering Talkers: Evaluations with Voiced Sentences and Normal-Hearing and Hearing-Impaired Listeners. *Journal of the Acoustical Society of America*, 87(1):359–372, January 1990.

[25] D. Graupe, J.K. Grosspietsch, and S.P. Basseas. A Single-Microphone-Based Self-Adaptive Filter of Noise from Speech and its Performance Evaluation. *Journal of Rehabilitation Research and Development*, 24(4):119–126, 1987.

[26] D.J. Van Tasell, S.Y. Larsen, and D.A. Fabry. Effects of an Adaptive Filter Hearing Aid on Speech Recognition in Noise by Hearing-Impaired Subjects. *Ear and Hearing*, 9(1):15–21, 1988.

[27] H. Levitt and J.C. Webster. *Effects of Noise and Reverberation on Speech*, chapter 16. Handbook of Acoustical Measurements and Noise Control. McGraw-Hill, 1991.

[28] C.M. Rankovic, R.L. Freyman, and P.M. Zurek. Potential Benefits of Adaptive Frequency-Gain Characteristics for Speech Reception in Noise. *Journal of the Acoustical Society of America*, 91:354–362, 1992.

[29] J.S. Lim and A.V. Oppenheim. Enhancement and Bandwith Compression of Noisy Speech. *Proceedings of the IEEE*, 67(12):1586–1604, December 1979.

[30] H.G. Hirsch. Intelligibility Improvement of Noisy Speech for People with Cochlear Implants. *Speech Communication*, 12:261–266, 1993.

[31] B.P. Milner and S.V. Vaseghi. Comparison of Some Noise-Compensation Methods for Speech Recognition in Adverse Environments. *IEE Proceedings Vision Image and Signal Processing*, 141(5), October 1994.

[32] J. Hardwick, C.D. Yoo, and J.S. Lim. Speech Enhancement Using the Dual Excitation Speech Model. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP-93*, volume 2, pages 367–370, 1993.

[33] J.F. Kaiser and E.E. David. Reproducing the Cocktail Party Effect. In *Journal of the Acoustical Society of America*, volume 32, page 918 (A), July 1960.

[34] J.F. Kaiser. Normalized Sound Control System. United States Patent Office, October 9, 1962. Patent No. 3,057,960.

[35] M.V.C. McConnell. A Two-Microphone Speech Enhancement System for Monaural Listening. Master's thesis, Massachusetts Institute of Technology, Cambridge, USA, Research Lab of Electronics, June 1985.

[36] B. Widrow et al. Adaptive Noise Cancelling: Principles and Applications. *Proceedings of the IEEE*, 63(12):1692–1719, December 1975.

[37] H.W. Strube. Separation of Several Speakers Recorded by Two Microphones (Cocktail-Party Processing). *Signal Processing*, 3(4):355–364, October 1981.

[38] P. Peterson, N. Durlach, W. Rabinowitz, and P. Zurek. Multimicrophone Adaptive Beamforming for Interference Reduction in Hearing Aids. *Journal of Rehabilitation Research and Development*, 24(2):103–110, 1987.

[39] D. Van Compernolle, W. Ma, F. Xie, and M. Van Diest. Speech Recognition in Noisy Environments with the Aid of Microphone Arrays. *Signal Processing*, 9(5/6):433–442, December 1990.

[40] A. Farassopoulos. *Speech Enhancement for Hearing Aids Using Real Time Adaptive Filtering Techniques.* PhD thesis, Departement D'Electricité, EPFL, Lausanne, Switzerland, 1992.

[41] J.E. Greenberg and P.M. Zurek. Evaluation of an Adaptive Beamforming Method for Hearing Aids. *Journal of the Acoustical Society of America*, 91(3):1662–1676, 1992.

[42] M. Kompis. *Der Adaptive Beamformer: Evaluation eines Verfahrens zur Störgeräuschunterdrückung für Hörgeräte.* PhD thesis, Institut für Biomedizinische Technik, ETHZ, Zürich, Switzerland, 1993.

[43] R.W. Stadler and W.M. Rabinowitz. On the Potential of Fixed Arrays for Hearing Aids. *Journal of the Acoustical Society of America*, 94(3):1332–1342, September 1993.

[44] O.M. Mitchell, C.A. Ross, and G.H. Yates. Signal Processing for a Cocktail Party Effect. *Journal of the Acoustical Society of America*, 50(2):656–660, May 1971.

[45] A. Souloumiac, P. Chevalier, and C. Demeure. Improvement in Non-Gaussian Jammer Rejection with a Non-Linear Spatial Filter. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP-93*, volume 5, pages 670–673, 1993.

[46] W. Knecht, R. Steiner, M. Joho, and G.S. Moschytz. Cancelling Spatial Interference with Nonlinear Filters. In H. Dedieu, editor, *European Conference on Circuit Theory and Design*, volume 1, pages 537–542. Elsevier, August/September 1993.

[47] W. Knecht, M. Schenkel, and G.S. Moschytz. Nonlinear Filters for Noise Reduction. In M.J.J. Holt et al., editor, *Signal Processing VII - Theories and Applications*, volume 3, pages 1500–1503, Edinburgh, Scotland, September 1994.

[48] W. Knecht. Nonlinear Noise Filtering and Beamforming Using the Perceptron and its Volterra Approximation. *IEEE Trans. Speech and Audio Processing*, 2(1):55–62, January 1994.

[49] B.D. Van Veen and K.M. Buckley. Beamforming: A Versatile Approach to Spatial Filtering. *IEEE ASSP Magazine*, pages 4–24, April 1988.

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

[51] M. Yanagida, Y. Miyoshi, Y. Nomura, and O. Kakusho. Application of the Least-Squares Method to Sound-Source Separation in a Multi-Source Environment. *Acustica*, 57:158–167, 1985.

[52] C. Jutten and J. Herault. Blind Separation of Sources, part 1: An Adaptive Algorithm based on Neuromimetic Architecture. *Signal Processing*, 24(1):1–10, July 1991.

[53] M.H. Cohen, P.O. Pouliquen, and A.G. Andreou. Analog LSI Implementation of an Auto-Adaptive Network for Real-Time Separation of Independent Signals. In J. Moody et al., editor, *Neural Information Processing Systems 4*, pages 805–812. Morgan Kaufmann, 1992.

[54] J.C. Platt and F. Faggin. Networks for the Separation of Sources that are Superimposed and Delayed. In J. Moody et al., editor, *Neural Information Processing Systems 4*, pages 730–737. Morgan Kaufmann, 1992.

[55] H.L. Nguyen, C. Jutten, and J. Caelen. Speech Enhancement: Analysis and Comparison of Methods on Various Real Situations. In J. Vandevalle et al., editor, *Signal Processing VI: Theories and Application*, pages 303–306. Elsevier, August 1992.

[56] M.J. Al-Kindi and J. Dunlop. Improved Adaptive Noise Cancellation in the Presence of Signal Leakage on the Noise Reference Channel. *Signal Processing*, 17(3):241–250, 1989.

[57] E. Weinstein, M. Feder, and A.V. Oppenheim. Multi-Channel Signal Separation by Decorrelation. *IEEE Trans. Speech and Audio Processing*, 1(4):405–413, October 1993.

[58] S. Van Gerven, D. Van Compernolle, H.L. Nguyen-Thi, and C. Jutten. Blind Separation of Sources: A Comparative Study of a 2-nd and a 4-th Order Solution. In M.J.J. Holt et al., editor, *Signal Processing VII - Theories and Applications*, volume 3, pages 1153–1156, Edinburgh, Scotland, September 1994.

[59] W. Soede. *Improvement of Speech Intelligibility in Noise.* PhD thesis, Delft University of Technology, 1990.

[60] R.W. Stadler. Optimally Directive Microphones for Hearing Aids. Master's thesis, Research Lab of Electronics, Massachusetts Institute of Technology, Cambridge, USA, September 1992.

[61] R.L. Pritchard. Maximum Directivity Index of a Linear Point Array. *Journal of the Acoustical Society of America*, 26(6):1034–1039, 1954.

[62] H. Cox, R.M. Zeskind, and T. Kooij. Practical Supergain. *IEEE Trans. Acoustics, Speech and Signal Processing*, 34(3):393–398, June 1986.

[63] P.M. Peterson. Simulating the Response of Multiple Microphones to a Single Acoustic Source in a Reverberant Room. *Journal of the Acoustical Society of America*, 80(5):1527–1529, November 1986.

[64] J.E. Greenberg and P.M. Zurek. Preventing Reverberation-Induced Target Cancellation in Adaptive-Array Hearing Aids. volume 95, pages 2990–2991. Journal of the Acoustical Society of America, May 1994.

[65] H.K. Kwan and Q.P. Li. New Nonlinear Adaptive FIR Digital Filter for Broadband Noise Cancellation. *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, 41(5):355–360, May 1994.

[66] G.L. Sicuranza. Quadratic Filters for Signal Processing. *Proceedings of the IEEE*, 80(8):1263–1285, August 1992.

[67] V.J. Mathews. Adaptive Polynomial Filters. *IEEE Signal Processing Magazine*, July 1991.

[68] M.J. Korenberg. Parallel Cascade Identification and Kernel Estimation for Nonlinear System. *Annals of Biomedical Engineering*, 19:429–455, 1991.

[69] K.S. Narendra and K. Parthasarathy. Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Trans. on Neural Networks*, 1:4–27, March 1990.

[70] E.M. Azoff. Reducing Error in Neural Networks Time Series Forecasting. *Neural Computing & Applications*, 1:240–247, 1993.

[71] D.R. Hush and B.G. Horne. Progress in Supervised Neural Networks. *IEEE Signal Processing Magazine*, pages 8–38, January 1993.

[72] S. Haykin. *Neural Networks: A Comprehensive Foundation.* Macmillan, 1994.

[73] K. Funahashi. On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks*, 2:183–192, 1989.

[74] R. Hecht-Nielsen. The Munificence of High Dimensionality. In I. Aleksander and J. Taylor, editors, *Artificial Neural Networks*, volume 2, pages 1017–1030. Elsevier Science Publishers B.V., 1992.

[75] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. *Learning Internal Representations by Error Propagation*, volume 1, chapter 8, pages 318–362. MIT Press, 1986.

[76] M. Møller. Supervised Learning on Large Redundant Training Sets. In *Neural Networks for Signal Processing*, pages 79–89. IEEE Press, 1992.

[77] U.A. Müller. *Simulation of Neural Networks on Parallel Computers*, volume 23 of *Series in Micro-Electronics.* Hartung-Gorre, Konstanz, 1993.

[78] Y. LeCun. Efficient learning and Second-Order Methods. In *Neural Information Processing Systems NIPS-93*, A Tutorial at NIPS-93, November 1993.

[79] J. Moody and C.J. Darken. Fast Learning in Networks of Locally-Tuned Processing Units. *Neural Computation*, 1:281–294, 1989.

[80] L. Tarassenko and S. Roberts. Supervised and Unsupervised Learning in Radial Basis Function Classifiers. *IEE Proc. Vision, Image Signal Processing*, 141(4):210–216, August 1994.

[81] J.N. Lin and R. Unbehauen. Adaptive Nonlinear Digital Filter with Canonical Piecewise Linear Structure. *IEEE Trans. on Circuits and Systems*, 37(3):347–353, March 1990.

[82] J.C. Stapleton and S.C. Bass. Adaptive Noise Cancellation for a Class of Nonlinear, Dynamic Reference Channels. *IEEE Trans. on Circuits and Systems*, 32(2):143–150, February 1985.

[83] S.W. Piché. Steepest Descent Algorithms for Neural Network Controllers and Filters. *IEEE Trans. on Neural Networks*, 5(2):198–212, March 1994.

[84] R. Steiner and M. Joho. Adaptive Störgeräuschunterdrückung mit Nichtlinearen Filtern. Master's thesis, ETH Zürich, Institut für Signal and Informationsverarbeitung, January 1993.

[85] S. Haykin. *Adaptive Filter Theory.* Prentice-Hall, second edition, 1991.

[86] D. Nguyen and B. Widrow. Improving the Learning Speed of 2-Layer Neural Networks by Choosing Initial Values of the Adaptive Weights. In *IJCNN International Joint Conference on Neural Networks*, volume III, pages 21–26. IEEE Publishing Services, June 1990.

[87] M. Giulieri. Perceptron mit Adaptiven Lernraten. Master's thesis, ETH Zürich, Institut für Signal and Informationsverarbeitung, July 1994.

[88] S.T. Alexander. *Adaptive Signal Processing.* Texts and Monographs in Computer Science. Springer Verlag, 1986.

[89] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1988.

[90] J.E. Greenberg, P.M. Peterson, and P.M. Zurek. Intelligibility-Weighted Measures of Speech-to-Interference Ratio and Speech System Performance. *Journal of the Acoustical Society of America*, 94(5):3009–3010, November 1993.

[91] B. Mulgrew. Orthonormal Functions for Nonlinear Signal Processing and Adaptive Filtering. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP-94*, volume 3, pages 509–512, April 1994.

[92] T.A. Kütükçüoglu. Alternative Adaptation Methods for Perceptrons. Semesterarbeit, Institut für Signal und Informationsverarbeitung, ETH Zürich, Switzerland, February 1994.

Seite Leer /
Blank leaf

Curriculum Vitae

Wolfgang G. Knecht

1962	Born on January 10 in Cologne, Germany.
1968–1972	Grundschule in Bergisch-Gladbach, Germany.
1972–1981	Otto-Hahn Gymnasium in Bergisch-Gladbach, Germany.
1982–1984	Vordiplom in Physics, University of Cologne, Germany
1985–1988	Diplom in Physics, Philipps University of Marburg, Germany.
1988–1991	Research Assistent at the Research Lab of Electronics, Massachusetts Institute of Technology, Cambridge, USA.
1991	Masters of Science in Physics, Massachusetts Institute of Technology, Cambridge, USA.
1991-1992	Teaching Assistant at the Signal and Information Processing Lab, ETH Zürich, Switzerland.
1992–1995	Research Assistant at the Signal and Information Processing Lab, ETH Zürich, Switzerland.
1995	PhD in Technical Sciences, ETH Zürich, Switzerland.
1995–	Development Engineer at Phonak AG, Stäfa, Switzerland