Nonlinear Functions for Blind Separation and Equalization

A dissertation submitted to the
Swiss Federal Institute of Technology, Zürich
for the degree of
Doctor of Technical Sciences

presented by

Heinz Mathis
dipl. El.-Ing. ETH
born on March 31, 1968
citizen of Küblis GR

accepted on the recommendation of
Prof. Dr. Hans-Andrea Loeliger, examiner
Prof. Dr. Scott C. Douglas, co-examiner

Hartung-Gorre Verlag, Konstanz, November 2001
Die Liebe ist langmütig, die Liebe ist gürtig. Sie ereifert sich nicht, sie prahlt nicht, sie bläht sich nicht auf. Sie handelt nicht ungehörig, sucht nicht ihren Vorteil, lässt sich nicht zum Zorne reizen, trägt das Böse nicht nach. Sie freut sich nicht über das Unrecht, sondern freut sich an der Wahrheit. Sie erträgt alles, glaubt alles, hofft alles, hält allem stand. Die Liebe hört niemals auf.

1. Korinther 13,4–8
I would like to thank Prof. G. S. Moschytz for providing such an excellent research environment with an academic freedom second to none and for inviting me to join his group. I would also like to thank his successor, Prof. H.-A. Loeliger, for acting as the referee of this thesis and for letting me continue with my ideas. Furthermore, I thank Prof. S. C. Douglas for acting as a co-referee and for hosting me as a guest researcher at the Southern Methodist University in Dallas during Summer 2001.

The signal and information processing laboratory (ISI) is a fantastic group to be a member of. I learned a lot from many different people and would like to thank them all for sharing their experience with me. The biggest thanks I owe to Marcel Joho, who got me acquainted with the subject of blind separation and with whom I shared many interests, largely due to the similar pace at which our families were growing. But my thanks also goes to Thomas von Hoff, whose publications on the stability analysis greatly influenced the direction of this thesis, to Pascal Vontobel for solving many a mathematical puzzle, to Hanspeter Schmid, Felix Lustenberger, and Stefan Moser for working out solutions to my Latex problems, and to Max Dünki for providing a stable Unix environment.

I would like to thank all the people who ran with me at lunch-times, in particular Markus Helfenstein, who really pushed me up that hill. I am also thankful to people with whom I could discuss non-work-related problems, in particular to Sigi Wyrsch and Dani Lippuner.

The people who proof-read this thesis have earned special acknowledgment for their endurance: Elsabe van den Heever, and again Marcel Joho and Thomas von Hoff.

My deepest thanks, however, goes to my family. I thank my parents for their constant support and encouragement throughout all the stages of my education.
I also thankfully remember the times I spent studying up in the Alps with my lovable Grandmother, Nani, who sadly passed away during the time I wrote up this thesis. I am deeply grateful to my wife, Franziska, for all her love, patience, support, and encouragement, and to my two boys, Lars and Roman, for simply being there.

Abstract

Nonlinear functions are an important part of blind adaptive algorithms solving filtering problems such as blind separation and blind equalization. Roughly speaking, they take over the role of a proper training reference signal, which is not available, hence the term “blind”.

The common idea shared by stochastic gradient-search algorithms either separating or deconvolving signals (or both) is to cross-correlate the signals before and after a nonlinear function, which reveals any existing higher-order correlation among the signals or among different time lags of the same signal. Such higher-order correlations indicate mutual dependence, which is then formed into an error signal to drive the output signals toward a state of higher independence. The underlying higher-order statistics are implicitly produced by the nonlinear functions. These nonlinear functions are essentially defined by the probability density function of the original source signals to extract and the cost function (such as mutual independence, maximum-likelihood, etc.).

In cases where the original distributions are unknown, change over time, or are of different nature, the nonlinearity has to adapt itself according to some estimate of the distribution, or be robust enough to cover a wide mismatch of the assumed model. Stability analyses reveal the stable regions of nonlinearities by determining the set of distributions for which a given nonlinearity results in local separation convergence. Unfortunately, no fixed nonlinearity can cover the entire universe of distributions.

Although the exact form of the nonlinearity might not matter for an algorithm to converge, it may have an impact on the convergence time of the separation/deconvolution process. By investigating these stability and performance parameters, robust, optimal, and universal (parametric) nonlinearities can be found. If complexity is an issue, simple nonlinearities are preferable.
to nonlinearities employing hyperbolic or polynomial functions. The threshold nonlinearity is such a simple nonlinearity. It works directly for sub-Gaussian signals such as typically used in digital data communications. Moreover, by adjusting the threshold, it may be used to separate any non-Gaussian signal.

Bias removal techniques, which remove any coefficient bias due to additive noise, are applicable to the threshold nonlinearity. The threshold nonlinearity is also well suited to blind equalization and blind carrier phase synchronization, a technique that is related to blind signal separation, and can hence be described within a common framework.

On the other hand, many simple algorithms for blind deconvolution, such as Sato’s algorithm and the constant-modulus algorithm (CMA) can be extended to work for a wider class of distributions by adding a simple coefficient norm factor in the update equation.

**Keywords:** Blind signal separation, blind equalization, blind carrier phase synchronization, nonlinear functions, threshold nonlinearity.

Techniken, die wegen additivem Rauschen eingehandelte Verschiebungen der Lösung (Bias) korrigieren, lassen sich kombinieren mit der Schwellwert-Nichtlinearität. Die Schwellwert-Nichtlinearität eignet sich ebenfalls für blinde Egalisation und für blinde Trägerphasensynchronisation, eine Technik die eng verwandt ist mit blinder Quellenseparierung und daher mit ähnlicher Notation beschrieben werden kann.

Auf der anderen Seite können viele einfache Algorithmen für die blinde Egalisation, wie z.B. Sato's Algorithmus oder der Constant-Modulus Algorithmus, mittels Hinzufügen eines Koeffizienten normterms in der Aufdatierungs-gleichung auf eine breitere Klasse von Verteilungen angewendet werden.

**Stichworte:** Blinde Quellenseparation, blinde Kanalegalisation, blinde Trägerphasensynchronisation, nichtlineare Funktionen, Schwellwert-Nichtlinearität.
### 3 Problem formulation

- 3.1 Model and assumptions ............................................. 40
- 3.2 Performance measures ............................................. 43
- 3.3 Narrowband model .................................................. 43
- 3.4 Relationship to beamforming ...................................... 44

### 3.2 Maximum-likelihood solution

- 3.2.1 Natural/relative gradient ...................................... 47

### 3.3 Accelerated update equation

- 3.3.1 Alternative derivation of the gradient modification ......... 48

### 3.4 Convergence behavior

- 3.5 Equilibrium points .................................................. 51

### 3.10 Summary ........................................................... 71

### 4 Nonlinear functions

- 4.1 Introduction .......................................................... 74
- 4.2 Gaussian sources ..................................................... 74
- 4.3 Scaling the nonlinearity ............................................ 75
- 4.4 Convergence speed of the BSS algorithm ...................... 77
- 4.5 Monomial functions ................................................... 79
- 4.6 Optimal nonlinearities .............................................. 82
- 4.7 A proof of the nonexistence of a universal nonlinearity .... 85

### 5 The Threshold Nonlinearity

#### 5.1 Formal derivation of the score function for the uniform distribution

- 5.1.1 The form of the nonlinearity .................................. 82

### 6 Blind equalization/deconvolution

#### 6.1 Introduction

- 6.1.1 Motivation .......................................................... 138

### 6.2 Problem formulation

- 6.2.1 Model and assumptions ........................................... 140
- 6.2.2 Performance measure ............................................. 141

### 6.3 Traditional Bussgang-type algorithms

- 6.3.1 The Sato algorithm ............................................... 142
- 6.3.2 Godard’s algorithm ................................................ 143
- 6.3.3 The constant-modulus algorithm (CMA) ...................... 143

### 6.4 The Bussgang algorithm

- 6.4.1 Uniformly distributed signals ................................ 141
- 6.4.2 M-QAM signals ..................................................... 141

### 6.5 Stabilization of arbitrary distributions

- 6.5.1 Known methods ................................................... 119
- 6.5.2 The adaptive threshold nonlinearity ......................... 122
- 6.5.3 Output normalization ............................................. 122
- 6.5.4 Advantages of the new method ................................ 126
- 6.5.5 Computer simulations ............................................ 126

### 6.6 Optimal nonlinearities

- 6.6.1 The form of the nonlinearity .................................. 82
- 6.6.2 Optimization of the nonlinearity .............................. 83

### 6.7 A proof of the nonexistence of a universal nonlinearity

- 6.7.1 The polynomial nonlinearity .................................. 85

### 6.8 Stabilizing blind separation algorithms

- 6.8.1 Summary of algorithms ........................................ 146
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.4</td>
<td>Increased convergence speed</td>
<td>146</td>
</tr>
<tr>
<td>6.5</td>
<td>An extension to Bussgang-type algorithms</td>
<td>149</td>
</tr>
<tr>
<td>6.5.1</td>
<td>Introduction</td>
<td>149</td>
</tr>
<tr>
<td>6.5.2</td>
<td>Local stability analysis</td>
<td>149</td>
</tr>
<tr>
<td>6.5.3</td>
<td>Stability for generalized Gaussian distributions</td>
<td>156</td>
</tr>
<tr>
<td>6.5.4</td>
<td>Algorithm modifications</td>
<td>157</td>
</tr>
<tr>
<td>6.5.5</td>
<td>Examples</td>
<td>160</td>
</tr>
<tr>
<td>6.5.6</td>
<td>Computer simulations</td>
<td>163</td>
</tr>
<tr>
<td>6.6</td>
<td>Natural-gradient updates</td>
<td>166</td>
</tr>
<tr>
<td>6.6.1</td>
<td>The threshold nonlinearity</td>
<td>166</td>
</tr>
<tr>
<td>6.7</td>
<td>Summary</td>
<td>167</td>
</tr>
<tr>
<td>7</td>
<td>Blind carrier phase synchronization</td>
<td>169</td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
<td>170</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Vestigial sideband modulation</td>
<td>170</td>
</tr>
<tr>
<td>7.2</td>
<td>Phase-error induced symbol error rates</td>
<td>171</td>
</tr>
<tr>
<td>7.2.1</td>
<td>SER of PAM with phase error</td>
<td>171</td>
</tr>
<tr>
<td>7.2.2</td>
<td>SER of QAM with phase error</td>
<td>172</td>
</tr>
<tr>
<td>7.2.3</td>
<td>SER of VSB with phase error</td>
<td>172</td>
</tr>
<tr>
<td>7.2.4</td>
<td>Symbol error floor</td>
<td>176</td>
</tr>
<tr>
<td>7.3</td>
<td>Residual phase-error correction</td>
<td>176</td>
</tr>
<tr>
<td>7.3.1</td>
<td>Data-aided phase-error correction</td>
<td>177</td>
</tr>
<tr>
<td>7.3.2</td>
<td>Blind phase synchronization</td>
<td>177</td>
</tr>
<tr>
<td>7.3.3</td>
<td>Blind phase-error correction for 8-VSB system</td>
<td>181</td>
</tr>
<tr>
<td>7.4</td>
<td>Simulation results</td>
<td>185</td>
</tr>
<tr>
<td>7.5</td>
<td>Summary</td>
<td>188</td>
</tr>
<tr>
<td>8</td>
<td>Concluding remarks</td>
<td>189</td>
</tr>
<tr>
<td>8.1</td>
<td>Conclusions</td>
<td>189</td>
</tr>
<tr>
<td>8.2</td>
<td>Outlook</td>
<td>190</td>
</tr>
<tr>
<td>A</td>
<td>Functions and formulae</td>
<td>193</td>
</tr>
<tr>
<td>A.1</td>
<td>The Gamma function</td>
<td>193</td>
</tr>
<tr>
<td>A.2</td>
<td>Stirling’s formula</td>
<td>194</td>
</tr>
<tr>
<td>A.3</td>
<td>Matrix-Inversion Lemma</td>
<td>194</td>
</tr>
<tr>
<td>A.4</td>
<td>Differentiation with respect to a matrix of complex elements</td>
<td>195</td>
</tr>
<tr>
<td>A.5</td>
<td>Derivation of the relationship between the fourth-order cumulant and higher-order moments</td>
<td>196</td>
</tr>
<tr>
<td>B</td>
<td>Constellations and SERs of some digital modulation formats</td>
<td>197</td>
</tr>
<tr>
<td>C</td>
<td>Formal proofs</td>
<td>199</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Motivation

Situations where information signals occur corrupted either by time-shifted replicas of their own signal or by other signals of similar nature can be found in the real world in a vast variety. Reverberation of acoustic signals in closed rooms, multipath propagation in wireless communications, blurred images, low-pass-filtered biomedical signals, to name just a few, are all examples illustrating the prevalence of unwanted convolution in different areas. Similarly, the problem of simultaneous excitations of multiple sources has applications in acoustics, communications, biomedical engineering, image processing, and even data mining (extraction of information or knowledge from data bases). Whereas the human brain can handle the so-called “cocktail-party effect” quite successfully, engineers have battled to devise algorithms resolving such situations.

In the wireless-communication world, recent government auctions of frequency bands have clearly indicated the need to make optimum use of the bandwidth available. Besides traditional approaches such as TDMA, FDMA, and CDMA, multiple use of common channels is possible using space division multiple access (SDMA), where the fact is exploited that no two transmitters or receivers have the same physical location. The basic configuration of a wireless communication system with several transmitters and receivers sharing the same channels can be visualized by Fig. 1.1.
Modern communication systems increasingly require training-less adaptation, either to save capacity, due to multipoint considerations in broadcast environments, or to accommodate unpredictable channel changes [135]. So-called self-recovering or blind algorithms are useful here, since they operate without the knowledge of some training sequence.

Although blind algorithms reach back as far as 30 years, it has only become an area of intense research activities over the last ten years. The new wave of research of blind algorithms has been motivated by two main factors. First, new fast-converging methods [7] have been discovered, based on information-theoretical concepts [12, 86]. Second, the semiconductor industry has developed new hardware systems that allow the real-time implementation of such algorithms.

1.2 Historical development

Historically, the problem of blind deconvolution appeared much earlier than blind separation. This is on the one hand due to suitable methods to blindly separate signals, which became available later than their counterpart for blind deconvolution, but on the other hand also due to a rising number of applications of MIMO (multiple in—multiple out) systems in recent years. Despite the fact that the two problems are related in nature and many applications require the simultaneous solution of both problems—so-called multichannel blind deconvolution (MCBD)—we separately list their respective historical development.

1.2.1 Blind deconvolution

The notion of blind deconvolution appeared for the first time in [121] and describes the problem of deconvolving two signals when both are unknown. Stockham’s method required the idea of homomorphic signal processing, a summary of which can be found in [107]. In short, homomorphic systems are nonlinear systems that obey a generalized principle of superposition. Homomorphic signal processing is particularly useful when two signals that are convolved are constitutionally different.

In the seventies and eighties, research into blind deconvolution was mainly driven by two areas. On the one hand the oil industry heavily subsidized projects investigating the analysis of geophysical layerings, which blind deconvolution was widely applied to, but with relatively few disclosed publications. On the other hand many pioneers in blind deconvolution had their roots in communications, which consequently was an area to which many original papers were devoted.

If decision-directed algorithms are regarded as blind algorithms—they do not assume knowledge of the input symbols, but estimate them—Lucky [91] certainly deserves to be mentioned as one of these pioneers. Very often, however, Sato [117] is considered the father of blind deconvolution algorithms in the form that represents the counterpart of the nonblind LMS stochastic-gradient update rule. At that time it was already clear that by using a nonlinear function, higher-order terms are implicitly generated, whose statistical independence are necessary in the most general case, rather than second-order independence or decorrelation only.

A very similar algorithm to Sato’s algorithm was devised independently by Godard [60] and by Treichler and Agee [126], and became subsequently known as either Godard’s algorithm in its more general case or the so-called constant-modulus algorithm (CMA), which is still used today.

The expression blind equalization, which has become a standard term in communications, was first used by Benveniste and Goursat [17]. They and other researchers [113] combined different error functions to get a smooth changeover from a blind to a decision-directed mode.

Alongside with the development of algorithms, there was a quest for sufficient and necessary conditions for blind equalization, so subsequently many results were of existential rather than constructive nature, i.e., they did not explicitly lead to the solutions, but provided the necessary framework in terms of...
conditions and assumptions, and increased the insight of solutions known at the
time and to follow later.

Among the first researchers to state sufficient conditions for perfect equal-
ization were Benveniste et al. [18], who showed that the original signal is re-
trieved if the probability density function (pdf) of the deconvolved sequence
matches that of the original signal. Later, Shalvi and Weinstein [120] relaxed
this condition by showing that perfect equalization is achieved by maximizing
the output kurtosis.

1.2.2 Blind separation

Blind signal separation (BSS), also referred to as blind source separation, is a
much younger research topic than blind deconvolution. The conference publi-
cations by Hérault and Jutten [67] followed by their journal publication [77]
are usually regarded as the seminal papers in blind signal separation. The non-
linearities used in their approach were, however, found more heuristically than
with mathematical rigor. Furthermore, they used a recurrent neural network as
opposed to later approaches, which mostly concentrated on feed-forward net-
works.

Jutten and Hérault [77] also came up with the term Independent Compo-
nents Analysis (INCA), which later became more standard as Independent Com-
ponent Analysis (ICA) [36]. Subsequently, ICA has been used as an equivalent
of BSS, but strictly speaking, ICA is more a method to solve the BSS prob-
lem. With ICA, a paradigm evolved that let many researchers think about the
ture cost function to minimize. In this spirit, cost functions motivated by infor-
mation theory arose. The key papers of this concept were written by Bell and
Sejnowski [11, 12].

A further milestone in the development of fast blind separation algorithms
was the introduction of a modified gradient, which led to considerably shorter
convergence times. Amari [2] and Cardoso [22] independently derived solu-
tions that resulted in the same update equation for the adaptive algorithm, and
named their method natural gradient and relative gradient, respectively.

1.3 Contributions of the thesis

The main subjects of this thesis are blind adaptive algorithms, in particular for
signal separation, with an emphasis on suitable nonlinearities, which reveal
dependencies among different signal parts (either temporal or spatial), and help
drive such dependencies to zero.

Special focus is set on the following topics:

- A concise treatment of higher-order moments, in particular fourth-order
  statistics (kurtosis) with respect to their influence on the behavior of blind
  adaptive algorithms is provided. The influence of timing offsets of dig-
  itally modulated signals to their kurtoses is of special interest. Moment
  relations of a large class of distributions can be expressed with respect
  to the Gamma function. Using the convex property of the Gamma func-
  tion allows the derivation of many useful relationships of higher-order
  moments of those distributions.

- The thesis contains a systematic overview of current blind separation
  algorithms and a comparison of their respective convergence times.
  Whereas the literature usually states the stability conditions for real-
  valued random variables, the more general conditions for complex values
  are derived here.

- Unlike most approaches, which exclude noise, the true ML gradient-
  search update equation for an AWGN channel is derived and put into
  relation to the more famous noiseless case. Furthermore, the applica-
  tion of bias removal techniques to the threshold nonlinearity algorithms
  is investigated.

- Working nonlinearities are categorized according to their suitability and
  complexity. The score function is shown to maximize robustness against
  model mismatch. It was previously assumed that universal nonlinearities
  for all possible distributions do exist. It is shown in this thesis that any
  approach to find such a fixed nonlinearity is inherently flawed.

- A very attractive nonlinear function in the form of a threshold nonline-
  arity is introduced that has favorable features such as low computational
  complexity. By introducing a parameter into this three-level quantizer,
  a universal parametric nonlinearity is created, which separates any non-
  Gaussian distributions, a fact that is proven in this thesis. Stability criteria
  for continuous and discrete distributions are derived. Making the thresh-
  old parameter adaptive allows the devising of an algorithm to separate
unknown distributions. There are also distributions that the “classical” nonlinearities fail to separate, but where the threshold nonlinearity succeeds.

- Many of the properties of blind equalization can be deduced from those of blind signal separation. Furthermore, sufficient conditions for local stability of Bussgang-type blind equalizers are derived, and extensions of such equalizers, which are normally designed to work for sub-Gaussian signals, to more general distributions are presented.
- The intimate relationship of blind signal separation and blind phase synchronization is yet another topic. The application of blind phase synchronization to HDTV signals confirms its usefulness.

1.4 Outline of the thesis

In Chapter 2 we deal with the statistical prerequisites for blind adaptive signal processing. Besides definitions and characteristics of moments and cumulants, and the introduction of some parametric distributions, we derive some important features of the kurtosis.

The formulation of the blind signal separation problem is a substantial subject of Chapter 3. We derive an adaptive maximum-likelihood solution, which will accompany us throughout the thesis. Other criteria are shortly treated and the stability conditions are reviewed.

Chapter 4 deals with nonlinear functions as used in blind adaptive algorithms. Their relevance to different distributions is investigated, and their resulting convergence properties are compared. In this chapter we also show a new justification for the score function, and provide a proof of the nonexistence of universal fixed nonlinearities for the separation of all non-Gaussian distributions. Finally, we offer a list of possible solutions to stabilize blind separation algorithms.

The main result of this thesis is the construction of a simple, piece-wise constant nonlinearity with properties similar to those of more complicated functions. The so-called threshold nonlinearity is the main topic of Chapter 5. We formally derive this nonlinear function from the uniform distribution and indicate its suitability to other distributions by studying its stability regions. In this context, discrete distributions are of particular interest, as they have many potential applications in data communications. The most fundamental result of this chapter is the proof of the stability of the threshold nonlinearity for any non-Gaussian signal for a properly chosen threshold value. By adapting this value, we devise an algorithm that separates a mixture of distributions with different kurtosis signs. Other extensions of the algorithm mitigate bias effect due to AWGN on the transfer medium.

Blind equalization, a problem that is in fact older than blind separation, is the theme of Chapter 6. We give an overview of Bussgang-type algorithms, investigate their stability regions, and present a means of extending such algorithms to the class of super-Gaussian signals. The use of the threshold nonlinearity in natural-gradient algorithms for blind deconvolution rounds off this chapter.

Chapter 7 consists of a treatment of blind phase synchronization algorithms, which can be regarded as a special case of blind separation algorithms. Their usefulness is demonstrated at the example of HDTV signals.

Finally, Chapter 8 contains the conclusions and some outlook on possible further research activities in the area of this thesis. A collection of formulae used in this thesis and some formal proofs can be found in the Appendix.
Chapter 2

Statistical prerequisites

Statistical quantities play an important role in blind signal processing in the construction of algorithms, their control, and their performance evaluation. It seems therefore appropriate to collect the necessary tool set into an introductory chapter. We first note the definitions of moments and cumulants and in particular the most important cumulant in blind signal processing—the normalized fourth-order cumulant or in short the kurtosis—in Section 2.1. In Section 2.2 probability density functions of some sources used in blind problems are shortly introduced and their respective kurtoses are given. An important feature of moments that is later used in Chapter 5 is moment ordering, which is introduced in Section 2.3. Salient properties of the kurtosis, such as the minimum kurtosis, the sum kurtosis, the influence of non-ideal sampling time on the kurtosis, and SNR estimation by the sample kurtosis are the subjects of Section 2.4.

2.1 Statistical moments and cumulants

2.1.1 Definitions

In the following we shall give some important definitions with respect to nonlinear moments and cumulants of random variables. The \( n \)th moment of a random...
variable $x$ is given by

$$M_x^{(n)} \triangleq E \left\{ x^n \right\} = \int_{-\infty}^{\infty} x^n p_x(x) \, dx. \quad (2.1)$$

Often, the central moments are required, which are given by

$$\bar{M}_x^{(n)} \triangleq E \left\{ (x - m_x)^n \right\} = \int_{-\infty}^{\infty} (x - m_x)^n p_x(x) \, dx. \quad (2.2)$$

$m_x \triangleq M_x^{(1)}$ is the linear expectation of $x$, and for $m_x = 0$, the two definitions of the moments coincide. The moments can also be computed through the characteristic function, which is given by

$$\phi(t) \triangleq E \left\{ e^{jtx} \right\} = \int_{-\infty}^{\infty} e^{jtx} p_x(x) \, dx. \quad (2.3)$$

The $n$th moment of $x$ is then given by

$$M_x^{(n)} = (-j)^n \left. \frac{d^n \phi(t)}{dt^n} \right|_{t=0}. \quad (2.4)$$

Intimately related to the characteristic function is the cumulant generating function, given by

$$K(t) \triangleq \ln \phi(t) = \ln E \left\{ e^{jtx} \right\}. \quad (2.5)$$

The $n$th cumulant can be computed as

$$C_x^{(n)} = (-j)^n \left. \frac{d^n K(t)}{dt^n} \right|_{t=0}. \quad (2.6)$$

Relationships between $M_x^{(n)}$ and $C_x^{(n)}$ may be formed [103] and are particularly easy for zero-mean, symmetrically distributed random variables. In this case we have

$$C_x^{(1)} = 0, \quad (2.7)$$
$$C_x^{(2)} = M_x^{(2)}, \quad (2.8)$$
$$C_x^{(3)} = 0, \quad (2.9)$$
$$C_x^{(4)} = M_x^{(4)} - 3 \left( M_x^{(2)} \right)^2. \quad (2.10)$$

For a derivation of Eq. (2.10), see also Appendix A.5.

### 2.1.2 Kurtosis of a signal

The kurtosis, which is simply the normalized fourth-order cumulant, belongs to the more important quantities used to describe a statistical variable. Particularly in blind signal processing, kurtosis plays a significant role, but other areas such as SNR estimation can also benefit from some of the properties of the kurtosis. For the rest of this thesis except for Section 2.4.4 we assume i.i.d. signals. We therefore use the terms signal and variable interchangeably apart from in the aforementioned section. The kurtosis of a real random variable $x$ is essentially given by its normalized fourth-order central moment. Two different definitions are commonly used, namely

$$\kappa_4 \triangleq \frac{E \left\{ |x - m_x|^4 \right\}}{\sigma_x^4} \quad (2.11)$$

and

$$\kappa_4 \triangleq \frac{E \left\{ |x - m_x|^4 \right\}}{\sigma_x^4} - 3, \quad (2.12)$$

with $m_x$ and $\sigma_x^2$ denoting the mean and variance of the random variable $x$, respectively. We stick to the latter definition, since it exhibits a couple of interesting properties. With Eq. (2.12), the kurtosis of symmetrical distributions corresponds to the fourth-order cumulant, normalized by the squared variance, as can be seen by inspection of Eq. (2.10). Gaussian variables have a zero kurtosis (mesokurtic), so that the sign of the kurtosis determines sub- or super-Gaussianity. Haykin [66] writes that although perhaps in an abusive way, a random variable is said to be sub-Gaussian or super-Gaussian if the kurtosis is negative or positive, respectively. Super-Gaussian signals have a positive kurtosis, and are also called leptokurtic, while sub-Gaussian signals have a negative kurtosis, and are referred to as platykurtic. Although super- and sub-Gaussianity were originally reserved [18] for pdfs of the form $p(x) = K \exp(-g(x))$, for even $g(x)$, and were determined by the asymptotic behavior of $g'(x)/x$, the connection between Gaussianity and kurtosis has found wide acceptance in the converse direction, too. Note that as we shall later see, Gaussianity is a sufficient but not necessary condition for distributions to be mesokurtic. In the following, we assume zero-mean variables, i.e., $m_x = 0$.

For complex signals, different definitions of the kurtosis are in use [138]. One possible definition is the same as for real signals (cf. Eq. (2.12)). Another, often used definition for the kurtosis of a zero-mean complex random variable...
is given by \[ \kappa_4' = \frac{E \{ |x|^4 \} - 2 \left( E \{ |x|^2 \} \right)^2 - E \{ |x|^2 \}^2}{\sigma_x^4}. \]  \hspace{1cm} (2.13)

For distributions exhibiting some circular symmetry we have \( E \{ x^2 \} = 0 \), so that Eq. (2.13) turns into

\[ \kappa_4' = \frac{E \{ |x|^4 \}}{\sigma_x^4} - 2, \]  \hspace{1cm} (2.14)

which is just 1 higher than the value obtained by Eq. (2.12), i.e., \( \kappa_4' = \kappa_4 + 1 \). Strictly speaking, the true condition for \( E \{ x^2 \} = 0 \) is that the real and imaginary parts of the random variable are statistically uncorrelated and have the same second-order moment.

Many researchers have tried to find intuitive explanations of what kurtosis exactly indicates, similarly to how one explains variance as deviation from its mean. Two notions have arisen, peakedness combined with tailedness, and bimodality. However, to either concept counter-examples were found by Darlington [38] and Hildebrand [68], respectively. Still, for most distributions in use, both intuitive grasps show their validity. A high kurtosis usually implies strong peakedness combined with tailedness in the pdf, or sometimes also called lack of shoulder. A low kurtosis on the other hand indicates high bimodality. The relationship between the form of the pdf and the kurtosis sign has been further investigated in [58].

\section*{2.2 Source distributions}

\subsection*{2.2.1 Continuous distributions}

Beside the many given fixed distributions for specific physical sources, parametric source distributions, which cover a whole range of different models are frequently used, since they allow viability checking and comparison of algorithms under different circumstances.

A large family of signals can be modeled using the generalized Gaussian distribution [13], whose pdf is given by

\[ p_x(x) = \frac{\alpha}{2\beta \Gamma \left( \frac{1}{\alpha} \right)} e^{-\frac{\alpha}{\beta} \left( \frac{|x|}{\sigma_x} \right)^\alpha}, \]  \hspace{1cm} (2.15)

where the parameters \( \alpha \) and \( \beta \) determine the shape and the variance of the distribution, respectively. \( \Gamma(.) \) denotes the Gamma function, which is defined in Appendix A.1. Distributions of the form given by Eq. (2.15) with \( 0 < \alpha < 2 \) are referred to as super-Gaussian, those with \( \alpha > 2 \) as sub-Gaussian distributions. For a plot of the Gamma function and other details, see Appendix A.1. Special values of the parameter \( \alpha \) lead to some very common distributions, e.g., \( \alpha = 1 \) makes Eq. (2.15) the Laplacian distribution, \( \alpha = 2 \) the Gaussian, and \( \alpha = \infty \) the uniform. Table 2.1 shows the calculation of the moments as a function of the parameters.

| distribution | \( \alpha \) | \( \beta \) | \( p_x(x) \) | \( E \{ |x|^P \} \) |
|-------------|------------|------------|---------------|------------------|
| Laplacian   | \( \frac{\sqrt{\alpha}}{2} \sigma_x \) | \( \frac{1}{\sqrt{\alpha} \sigma_x} e^{-\frac{\alpha}{\sqrt{2 \beta}} \sqrt{2 \beta} \left( \frac{|x|}{\sigma_x} \right)^{\frac{\alpha}{\sqrt{2}}} \} \sqrt{2 \beta} \) | \( \Gamma\left( p + 1 \right) \left( \frac{\sqrt{\alpha}}{2} \sigma_x \right)^p \) |
| Gaussian    | \( \sqrt{2 \sigma_x} \) | \( \frac{1}{\sqrt{2 \sigma_x}} e^{-\frac{2}{\beta}} \frac{1}{\Gamma\left( \frac{1}{2} \right)} \cdot \frac{1}{\Gamma\left( \frac{1}{\alpha} \right)} \left( \frac{\alpha}{\sqrt{2} \sigma_x} \right)^p \) |
| uniform    | \( \infty \) | \( \sqrt{5 \sigma_x} \) | \( \frac{1}{\beta^{p+1}} \) | \( \frac{\Gamma\left( \frac{1}{2} \right)^p}{\Gamma\left( \frac{1}{\alpha} \right)} \) |
| general    | \( \frac{\Gamma\left( \frac{1}{2} \right) \Gamma\left( \frac{1}{\alpha} \right)}{\Gamma\left( \frac{1}{\alpha} \right)} \sigma_x \) | Eq. (2.15) | \( \frac{\Gamma\left( \frac{p+1}{\alpha} \right)}{\Gamma\left( \frac{1}{\alpha} \right)} \beta^p \) |

Table 2.1: Moments of the generalized Gaussian distribution.

For the generalized Gaussian distribution, the parameter \( \alpha \) determines the kurtosis in the following way [62]

\[ \kappa_4 = \frac{\Gamma\left( \frac{1}{\alpha} \right) \Gamma\left( \frac{1}{2} \right)}{\left( \Gamma\left( \frac{1}{\alpha} \right) \Gamma\left( \frac{1}{2} \right) \right)^2} - 3. \]  \hspace{1cm} (2.16)

The kurtosis of the generalized Gaussian distribution is not upper bounded for \( \alpha \to 0 \), but lower bounded by \(-1.2 \) for \( \alpha \to \infty \) (uniform distribution). Fig. 2.1
14 Chapter 2 Statistical prerequisites

(upper left) and Fig. 2.2 visualize the distribution given by Eq. (2.15) and the kurtosis given by Eq. (2.16), respectively.

Only unimodal signals can be expressed by the generalized Gaussian distribution given by Eq. (2.15). However, digital data signals are mostly bimodal or multimodal. The two-tailed Gamma distribution [68], whose pdf is

\[ p_x(x) = \frac{\beta^a}{2\Gamma(a)} |x|^{a-1} e^{-\beta |x|}, \quad -\infty < x < \infty, \]  

(2.17)

can model a wider range of distributions than the generalized Gaussian family. Some calculations involving this family, however, tend to be more difficult, due to the simultaneous appearance of a power of \( x \) and the exponential function in the density function. For \( \alpha = 1 \), Eq. (2.17) too expresses the Laplacian distribution. Pitfalls are lurking when handling the numerical representation of the pdf of the two-tailed Gamma distribution for extreme values of the parameter \( \alpha \). Using Stirling’s formula (see Appendix A.2), we can find an approximation of Eq. (2.17) for large values of \( \alpha \),

\[ p_x(x) = \frac{1}{2} \sqrt{\frac{\alpha}{2\pi}} e^{\alpha[(1-|x|+\ln|x|)-\ln|x|]}, \quad -\infty < x < \infty, \quad \alpha \gg 1, \]  

(2.18)

which avoids some numerical problems associated with large terms. Table 2.2 shows the calculation of the moments as a function of the parameters. The distribution and the kurtosis as a function of the parameter \( \alpha \) are again shown in Fig. 2.1 (upper right) and Fig. 2.2, respectively.

| distribution | \( \alpha \) | \( \beta \) | \( p_x(x) \) | \( E[|x|^p] \) |
|-------------|-----------|-----------|-------------|-------------|
| Laplacian   | 1         | \( \sqrt{2}/\sigma_x \) | \( -\frac{1}{\sqrt{2\pi\sigma_x}} e^{-x^2/2\sigma_x^2} \) | \( \Gamma(p+1) \left( \frac{\sigma_x^2}{2} \right)^p \) |
| binary      | \( \infty \) | \( \infty \) | \( \frac{\Gamma(\alpha)}{\Gamma(\alpha+1)} \left( \frac{\sigma_x^2}{2} \right)^{\alpha-1} \) | \( \frac{\Gamma(\alpha+p)}{\Gamma(\alpha+1)} \) |
| general     | \( \alpha \) | \( \sqrt{2}/\sigma_x \) | \( \sqrt{\frac{\alpha}{2\pi}} e^{\alpha[(1-|x|+\ln|x|)-\ln|x|]} \) | \( \frac{\Gamma(\alpha+p)}{\Gamma(\alpha+1)} \) |

Table 2.2: Moments of the two-tailed Gamma distribution.

The two-tailed Gamma distribution exhibits a kurtosis as given by

\[ \kappa_4 = \frac{(\alpha+3)(\alpha+2)}{(\alpha+1)^2} - 3. \]  

(2.19)

We see that this distribution can model signals with a kurtosis all the way down to \(-2\), hence unimodal and bimodal signals.

If only sub-Gaussian signals are of interest, a possible parametric distribution can be obtained from the symmetric Beta distribution [68, 116], which is given by

\[ p_x(x) = \frac{\Gamma(2\alpha)}{\Gamma^2(\alpha)} x^{\alpha-1}(1-x)^{\alpha-1}, \quad 0 < x < 1. \]  

(2.20)

Shifting and scaling of Eq. (2.20) leads to an unbiased, symmetric Beta distribution

\[ p_x(x) = \frac{\Gamma(2\alpha)}{\Gamma^2(\alpha)} \beta \cdot \frac{1}{\beta} \left( \frac{x}{\beta} + \frac{1}{2} \right)^{\alpha-1} \left( \frac{1 - x}{\beta} + \frac{1}{2} \right)^{\alpha-1}, \quad -\beta/2 < x < \beta/2, \]  

(2.21)

where the parameter \( \beta \) as with the other two pdfs depends on the variance of the signal, and is given by

\[ \beta = 2\sqrt{2\alpha + 1}\sigma_x. \]  

(2.22)
Chapter 2. Statistical prerequisites

For large $\alpha$, the fraction of two Gamma functions is numerically unstable, so using Stirling’s formula again we can rewrite Eq (2.21) into

$$p_x(x) = \frac{\Gamma(\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{\alpha-1} \left(1 - \frac{x}{\beta}\right)^{\alpha-1}}{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right) \frac{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{\alpha-1} \left(1 - \frac{x}{\beta}\right)^{\alpha-1}}{\Gamma(2\alpha) \cdot \frac{1}{\beta}}},$$

$-\beta/2 < x < \beta/2, \quad \alpha \gg 1. \quad (2.23)$

Since the moments of the symmetric Beta distribution $p_x(.)$ are given by $[116]

$$E[x^p] = \frac{\Gamma(\alpha + p) \cdot \Gamma(2\alpha)}{\Gamma(2\alpha + p)} \cdot \frac{\Gamma(\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{\alpha-1} \left(1 - \frac{x}{\beta}\right)^{\alpha-1}}{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right) \frac{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{\alpha-1} \left(1 - \frac{x}{\beta}\right)^{\alpha-1}}{\Gamma(2\alpha) \cdot \frac{1}{\beta}}}, \quad (2.24)$$

the moments of the unbiased, symmetric Beta distribution, which is again depicted in Fig. 2.1, are then given by

$$E[x^p] = E\{\beta(x - mx)^p\} = \beta^p \sum_{n=0}^{p} (-1)^n \binom{p}{n} E[x^n] m^{p-n} = \frac{\Gamma(2\alpha) \cdot \beta^p \sum_{n=0}^{p} (-1)^n \binom{p}{n} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{2\alpha-n} \Gamma(\alpha + n) \Gamma(2\alpha + n)}{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right) \frac{\Gamma(2\alpha) \cdot \frac{1}{\beta} \left(\frac{x}{\beta} + \frac{1}{2}\right)^{\alpha-1} \left(1 - \frac{x}{\beta}\right)^{\alpha-1}}{\Gamma(2\alpha) \cdot \frac{1}{\beta}}}, \quad (2.25)$$

The kurtosis of unbiased, symmetric Beta distributed signals is given by

$$\kappa_4 = -6/(2\alpha + 3), \quad (2.26)$$

hence reaches from zero (Gaussian) to $-2$ (binary). Its dependency on $\alpha$ is also depicted in Fig. 2.2.

Finally, the kurtosis of some other continuous distributions$^1$ of interest are

- $-1.5$ for a sine wave,
- $-1.2$ for a triangular or sawtooth signal (same as uniform distribution), and again $-2$ for a rectangular pulse signal (same as BPSK signal).

2.2.2 Discrete distributions

Baseband representations of sampled, digitally modulated signals have discrete distributions rather than continuous forms as treated in the previous section.

$^1$The following signals are deterministic ones, so that the term “distributions” refers to the amplitude density rather than the probability density.
Chapter 2. Statistical prerequisites

2.5.2 Source distributions

Their kurtoses can be calculated in closed form from their probability distributions using Eq. (2.12). The results are given in Table 2.3, and plots of the kurtoses for PAM (pulse amplitude modulation, also referred to as amplitude shift keying or short ASK) and QAM (quadrature amplitude modulation) are depicted in Fig. 2.2. Note that both definitions of the kurtosis of a complex signal (Eqs. (2.12) and (2.13)) are given in Table 2.3. An interesting point arises for M-PAM as M approaches infinity. As we steadily increase the number of possible values of the source variable and therefore the modality, the kurtosis approaches —1.2, the kurtosis for a uniform distribution, which is referred to as being nonmodal by Hildebrand [68].

For square QAM constellations, M needs to be a square of an integer. However, other constellations are possible, and are used in current technology, e.g., the voice modem standards V32 and V.33 use 32-QAM and 128-QAM, respectively. Because an odd power of two can always be written as

\[ M = 2^{2m+1} = 2^{2m-2} \cdot 2^3 = (2^{m-1})^2 \cdot (3^2 - 1) \]

or

\[ = (2^{m-1} \cdot 3)^2 - (2^{m-1})^2, \]

(2.27)
a difference of two squares, each of which is divisible by four for integer values of m \( \geq 2 \), a constellation can always be built where a large square is pruned by four small corner squares, e.g., for 32-QAM we skip the four corner points of a 36-point square (\( m = 2 \)). The exact kurtosis of such constellations, also referred to as cross constellations, lacks the easy form of square constellations but can still be expressed in closed form as given in row 3 of Table 2.3. A similar table containing the kurtoses of digitally modulated distributions can be found in [123].

An interesting case of a signal whose kurtosis does not monotonously increase with growing M is the variable-rate star QAM as suggested by Webb and Hanzo [133]. Their method of increasing the data rate is based on doubling alternately the number of amplitude rings and the number of points per ring in the constellation. For two rings, the two amplitudes are in the ratio of three-to-one. With four rings, the 1st (inner most) and 4th ring stay the same as with two rings, the other two amplitudes are chosen such that the Euclidean distances between the 2nd and 3rd, and between the 3rd and 4th ring are equal, and at the same time preserve the average energy from smaller constellations. The resulting amplitudes (power normalized) are

\[
\begin{align*}
\frac{1}{\sqrt{5}}, & \quad \frac{-2+2\sqrt{11}}{5\sqrt{5}}, \quad \frac{6+2\sqrt{41}}{5\sqrt{5}}, \quad \frac{3}{\sqrt{5}}
\end{align*}
\]

The kurtosis is an indicator of the convergence speed of blind adaptive algorithms. Due to higher absolute values of the kurtosis, it is easier to separate star-QAM signals with \( M = 2, 4, 32, 64 \) than with \( M = 8, 16 \). This is of course only true for blind methods that make use of the kurtosis as an independence measure. If decision-directed methods are employed, whose performance are directly related to the symbol error rate, fewer constellation points facilitate faster convergence times.

In order to have a parametric discrete distribution covering a wide range of the kurtosis, we introduce a ternary distribution with \( \Pr(x = 1) = \Pr(x = -1) = p \) and \( \Pr(x = 0) = 1 - 2p \). The kurtosis of this variable depends on \( p \) in the following way

\[ \kappa_4 = \frac{1}{2p} - 3. \]

(2.28)

This family of distributions covers the whole kurtosis range from \( -\infty \) to \(-2\) as \( p \) goes from 0 to \( \frac{1}{2} \). For \( p = \frac{1}{6} \) we get zero kurtosis. Although this choice of

Figure 2.2: Kurtoses of generalized Gaussian, two-tailed Gamma, and symmetric Beta distributed signals (all vs. \( \alpha \)), M-PAM signals (vs. \( M \)), and M-QAM signals (vs. \( M \)).

\[ 2\text{.5.2 Source distributions} \]

\[ \begin{align*}
\text{super-Gaussian} \\
\text{sub-Gaussian} \\
\text{generalized Gaussian} \\
\text{two-tailed Gamma} \\
\text{symmetric Beta} \\
\star \text{ M-PAM} \\
\star \text{ M-QAM}
\end{align*} \]

\[ \begin{align*}
\text{super-Gaussian} \\
\text{sub-Gaussian} \\
\text{generalized Gaussian} \\
\text{two-tailed Gamma} \\
\text{symmetric Beta} \\
\star \text{ M-PAM} \\
\star \text{ M-QAM}
\end{align*} \]

\[ \begin{align*}
\text{super-Gaussian} \\
\text{sub-Gaussian} \\
\text{generalized Gaussian} \\
\text{two-tailed Gamma} \\
\text{symmetric Beta} \\
\star \text{ M-PAM} \\
\star \text{ M-QAM}
\end{align*} \]
p makes the signal closest to a Gaussian distribution, other than fourth-order cumulants need not necessarily be zero.

### 2.2.3 Complex continuous distributions

As $M$ grows, the QAM distributions resemble more and more those of continuous ones, or at least modeling such discrete distributions by complex, continuous distributions becomes more accurate. Some important complex, continuous distributions are listed in Table 2.4.

<table>
<thead>
<tr>
<th>name</th>
<th>pdf plot</th>
<th>$\kappa_4$</th>
<th>$\kappa'_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>complex Gaussian</td>
<td><img src="image" alt="Complex Gaussian plot" /></td>
<td>$-1$</td>
<td>$0$</td>
</tr>
<tr>
<td>complex uniform</td>
<td><img src="image" alt="Complex Uniform plot" /></td>
<td>$-1.6$</td>
<td>$-0.6$</td>
</tr>
<tr>
<td>cyclic uniform</td>
<td><img src="image" alt="Cyclic Uniform plot" /></td>
<td>$-\frac{3}{3}$</td>
<td>$-\frac{2}{3}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>name</th>
<th>pdf plot</th>
<th>$\kappa_4$</th>
<th>$\kappa'_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>complex Gaussian</td>
<td><img src="image" alt="Complex Gaussian plot" /></td>
<td>$-1$</td>
<td>$0$</td>
</tr>
<tr>
<td>complex uniform</td>
<td><img src="image" alt="Complex Uniform plot" /></td>
<td>$-1.6$</td>
<td>$-0.6$</td>
</tr>
<tr>
<td>cyclic uniform</td>
<td><img src="image" alt="Cyclic Uniform plot" /></td>
<td>$-\frac{3}{3}$</td>
<td>$-\frac{2}{3}$</td>
</tr>
</tbody>
</table>

Table 2.4: Some complex, continuous distributions and their respective kurtoses.

### 2.3 Moment ordering

Consider $\tilde{n}$ a unit-variance Gaussian variable $\tilde{n} \sim \mathcal{N}(0,1)$ and $x$ the generalized Gaussian variable given by Eq. (2.15) parameterized by $\alpha$, which is larger (smaller) than 2 for sub- (super-) Gaussian signals. We will also consider a normalized version of $x$, called $\hat{x}$. The value of the parameter $\beta$ can be found from the general expression for the $m$th-order moment of a generalized Gaussian signal [84]

$$E \{ |x|^m \} = \frac{\Gamma \left( \frac{m+1}{\alpha} \right)}{\Gamma \left( \frac{1}{\alpha} \right)} \beta^m. \tag{2.29}$$

For $m = 2$, Eq. (2.29) gives

$$E \{ |x|^2 \} = \frac{\Gamma \left( \frac{3}{\alpha} \right)}{\Gamma \left( \frac{1}{\alpha} \right)} \beta^2 \tag{2.30}$$

or

$$\beta = \sqrt{\frac{\Gamma \left( \frac{1}{\alpha} \right)}{\Gamma \left( \frac{3}{\alpha} \right)}} \sigma_x \tag{2.31}$$

a result already used in Table 2.1. By using $\alpha = 2$ in Eqs. (2.29) and (2.30) we have the following relationship between the even moments of the unit-variance Gaussian variable

$$E \{ |\tilde{n}|^m \} = (m-1)E \{ |\tilde{n}|^{m-2} \}. \tag{2.32}$$

Similarly, for the even moments of generalized Gaussian variables with unit variance we can find

$$E \{ |\hat{x}|^m \} = \frac{\Gamma \left( \frac{m+1}{\alpha} \right)}{\Gamma \left( \frac{1}{\alpha} \right)} \frac{\Gamma \left( \frac{m+1}{\alpha} \right)}{\Gamma \left( \frac{3}{\alpha} \right)} \tag{2.33}$$

The moments of any unit-variance random variable build a monotonic increasing function in $m$

$$E \{ |\hat{x}|^m \} \geq E \{ |\hat{x}|^{m-2} \}, \quad m \geq 4. \tag{2.34}$$
with equality if and only if \( x \) is a binary random variable \( \in \{ \pm 1 \} \). A proof of Eq. (2.34) can be found in Appendix C.3. A related result to Eq. (2.34) is the corollary following the next lemma, which in turn was stated by Loève [90]. It will be used subsequently in Chapter 6.

**Lemma 2.1**

For the higher-order moments of a random variable \( x \) we note that

\[
E \left\{ |x|^m \right\} E \left\{ |x|^{b-a_1} \right\} \geq E \left\{ |x|^{a_2} \right\} E \left\{ |x|^{b-a_2} \right\}, \quad 0 < a_1 < a_2 \leq b/2.
\]

(2.35)

A proof of Lemma 2.1 can be found in [111, p. 343].

**Corollary 2.1**

The ratio \( \frac{E \left\{ |x|^{m+1} \right\}}{E \left\{ |x|^{m-1} \right\}} \) is a nondecreasing function in \( m \) for any distribution \( p_x(\cdot) \).

Proof: By setting \( b = 2m + \delta, a_1 = m - 1, a_2 = m + \delta - 1 \), with \( \delta \) being a small positive real number, \( 0 < \delta \leq 2 \), and rearranging Eq. (2.35) we get

\[
\frac{E \left\{ |x|^{m+\delta+1} \right\}}{E \left\{ |x|^{m+\delta-1} \right\}} \geq \frac{E \left\{ |x|^{m+1} \right\}}{E \left\{ |x|^{m-1} \right\}}.
\]

(2.36)

from which we conclude that \( E \left\{ |x|^{m+1} \right\} / E \left\{ |x|^{m-1} \right\} \) is a nondecreasing function in \( m \).

In fact, Eq. (2.34) can also be proved using Corollary 2.1 and noting that for \( m = 1 \) and a unit-variance random variable we have

\[
\frac{E \left\{ |\tilde{x}|^{m+1} \right\}}{E \left\{ |\tilde{x}|^{m-1} \right\}} \bigg|_{m=1} = E \left\{ |\tilde{x}|^2 \right\} = 1.
\]

(2.37)

2.3. Moment ordering

Furthermore, if \( \tilde{x} \) is super-Gaussian distributed, we find [1]

\[
E \left\{ |\tilde{x}|^m \right\} > E \left\{ |\hat{h}|^m \right\}, \quad m > 2,
\]

(2.38)

\[
E \left\{ |\tilde{x}|^m \right\} < E \left\{ |\hat{h}|^m \right\}, \quad m < 2.
\]

(2.39)

Vice versa, for \( \tilde{x} \) being sub-Gaussian distributed

\[
E \left\{ |\tilde{x}|^m \right\} < E \left\{ |\hat{h}|^m \right\}, \quad m > 2,
\]

(2.40)

\[
E \left\{ |\tilde{x}|^m \right\} > E \left\{ |\hat{h}|^m \right\}, \quad m < 2.
\]

(2.41)

Although stochastic ordering is also possible for fractional moments [1], we only consider even integers for \( m \) in what follows. The \( m \)th moments for \( m < 2 \) are given for completeness and will not be used further. In fact, using Eqs. (2.32) and (2.34) in Eqs. (2.38) and (2.40) we realize that

\[
E \left\{ |\tilde{x}|^m \right\} > (m - 1)E \left\{ |\hat{h}|^{m-2} \right\}, \quad x \text{ super-Gaussian distributed},
\]

(2.42)

\[
E \left\{ |\tilde{x}|^m \right\} < (m - 1)E \left\{ |\hat{h}|^{m-2} \right\}, \quad x \text{ sub-Gaussian distributed}.
\]

(2.43)

Tighter bounds can be obtained, namely

\[
E \left\{ |\tilde{x}|^m \right\} > (m - 1)E \left\{ |\hat{h}|^{m-2} \right\}, \quad x \text{ super-Gaussian distributed},
\]

(2.44)

\[
E \left\{ |\tilde{x}|^m \right\} < (m - 1)E \left\{ |\hat{h}|^{m-2} \right\}, \quad x \text{ sub-Gaussian distributed},
\]

(2.45)

where in the right-hand sides of Eqs. (2.42) and (2.43) \( \hat{h} \) has been replaced by \( \tilde{x} \). A proof of Eqs. (2.44) and (2.45) is provided in Appendix C.4.

Most of the equalities and inequalities from Eq. (2.32) to Eq. (2.45) can be extended to distributions whose variance is unequal to one by noting that

\[
E \left\{ |x|^m \right\} = \sigma_x^m E \left\{ |\tilde{x}|^m \right\}
\]

if \( \tilde{x} \) is the normalized version of \( x \). In particular, the moments of a Gaussian variable \( n \sim \mathcal{N}(0, \sigma_n^2) \) are related as

\[
E \left\{ |n|^m \right\} = (m - 1)E \left\{ |n|^{m-2} \right\} \sigma_n^2.
\]

(2.47)

The even moments of generalized Gaussian variables with variance \( \sigma_x^2 \) are related as

\[
E \left\{ |x|^m \right\} = \frac{\Gamma \left( \frac{m+1}{\alpha} \right)}{\Gamma \left( \frac{1}{\alpha} \right)} \cdot \frac{\Gamma \left( \frac{1}{\alpha} \right)}{\Gamma \left( \frac{m-1}{\alpha} \right)} \cdot \sigma_x^2.
\]

(2.48)
For arbitrary distributions we have
\[
E \{ |x|^m \} \geq E \{ |x|^{m-2} \} \frac{\sigma_x^2}{m} , \quad m \geq 4 , \quad (2.49)
\]
with equality if and only if \( x \) is binary distributed. And finally,
\[
E \{ |x|^m \} > (m-1)E \{ |x|^{m-2} \} \frac{\sigma_x^2}{m} , \quad x \) super-Gaussian distributed, \quad (2.50)
\]
\[
E \{ |x|^m \} < (m-1)E \{ |x|^{m-2} \} \frac{\sigma_x^2}{m} , \quad x \) sub-Gaussian distributed. \quad (2.51)
\]
Both the moments and the ratio of increasing moments are strictly monotonic increasing functions for the generalized Gaussian distribution as can be seen from Figs. 2.3 and 2.4, respectively. Moreover, \( E \{ |x|^m \} \) is a convex function in \( m \) for any distribution except for the binary distribution, where it is a constant, and the \( E \{ |x|^m \} / E \{ |x|^{m-2} \} \) is a convex function for super-Gaussian signals.

**Figure 2.3:** Moments for the generalized Gaussian distribution with unit variance.

**Figure 2.4:** Ratio of moments for the generalized Gaussian distribution with unit variance.

Another relationship between moments of the generalized Gaussian distribution will be used in Chapter 6 and bases on the following lemma.

**Lemma 2.2**

For \( 0 < \alpha < 2 \) and \( p < q \) the ratio of Gamma functions behaves as
\[
\frac{\Gamma \left( \frac{p+2}{\alpha} \right)}{p \Gamma \left( \frac{q+2}{\alpha} \right)} < \frac{\Gamma \left( \frac{q+2}{\alpha} \right)}{q \Gamma \left( \frac{p}{\alpha} \right)} . \quad (2.52)
\]

**Proof:** We have to show that for
\[
f(m) \triangleq \frac{\Gamma \left( \frac{m+2}{\alpha} \right)}{m \Gamma \left( \frac{m}{\alpha} \right)} \quad (2.53)
\]
we have
\[
f'(m) \triangleq \frac{df(m)}{dm} > 0 , \quad 0 < \alpha < 2 . \quad (2.54)
\]
We define
\[ \delta \triangleq \frac{2 - \alpha}{\alpha}, \quad 0 < \alpha < 2 \] (2.55)
and make use of the recursive property of the Gamma function to note that
\[ \Gamma \left( \frac{m + 2}{\alpha} \right) = \Gamma \left( \frac{m}{\alpha} + \delta + 1 \right) = \left( \frac{m}{\alpha} + \delta \right) \Gamma \left( \frac{m}{\alpha} + \delta \right). \] (2.56)

Using Eq. (2.56) in Eq. (2.53) and taking the derivative with respect to \( m \) leads to
\[ f'(m) = \frac{\left( \Gamma \left( \frac{m}{\alpha} + \delta \right) \Gamma \left( \frac{m}{\alpha} + \delta \right) \right)^{\prime}}{m \Gamma \left( \frac{m}{\alpha} \right)} \]
\[ = \frac{\left( \Gamma \left( \frac{m}{\alpha} + \delta \right) \Gamma' \left( \frac{m}{\alpha} + \delta \right) \right) \Gamma \left( \frac{m}{\alpha} + \delta \right) + \left( \Gamma \left( \frac{m}{\alpha} + \delta \right) \Gamma \left( \frac{m}{\alpha} + \delta \right) \right) \Gamma' \left( \frac{m}{\alpha} + \delta \right)}{m \Gamma \left( \frac{m}{\alpha} \right)} \]
\[ - \frac{\left( \Gamma \left( \frac{m}{\alpha} + \delta \right) \Gamma' \left( \frac{m}{\alpha} + \delta \right) \right) \Gamma' \left( \frac{m}{\alpha} + \delta \right)}{m \Gamma \left( \frac{m}{\alpha} \right)} \] (2.57)

Using (cf. [9])
\[ \Gamma'(x) = \xi(x) \Gamma(x) \] (2.58)
with
\[ \xi(x) \triangleq -C - \frac{1}{x} + \sum_{n=1}^{\infty} \frac{x}{m(x+n)}, \] (2.59)

where \( C \) is Euler’s constant, we can write
\[ f'(m) = \frac{\Gamma \left( \frac{m}{\alpha} + \delta \right) \Gamma \left( \frac{m}{\alpha} \right)}{m \Gamma \left( \frac{m}{\alpha} \right)} \left( \xi \left( \frac{m}{\alpha} + \delta \right) - \xi \left( \frac{m}{\alpha} \right) \right) \] (2.60)

which leads after some calculation and using Eq. (2.59) to
\[ f'(m) = \frac{\Gamma \left( \frac{m}{\alpha} + \delta \right)}{m \Gamma \left( \frac{m}{\alpha} \right)} \frac{m^2 + \delta \alpha m}{\alpha^2} \left( \sum_{n=1}^{\infty} n \left( \frac{m}{\alpha} + \delta + n \right) - \sum_{n=1}^{\infty} n \left( \frac{m}{\alpha} + n \right) \right). \] (2.61)

The first two factors on the RHS of Eq. (2.61) are positive for \( 0 < \alpha < 2 \), and the difference of the two sums is
\[ \sum_{n=1}^{\infty} \frac{m}{\alpha} + \delta + n - \sum_{n=1}^{\infty} \frac{m}{\alpha} + n = \sum_{n=1}^{\infty} \frac{\delta}{\alpha} \frac{m}{\alpha} + n > 0, \] (2.62)
making \( f'(m) \) positive for all \( m > 0 \). \( \square \)

The converse of Lemma 2.2, together with Lemma 2.2 itself, will be used in Chapter 3 and is presented in the following corollary.

**Corollary 2.2**
For \( \alpha > 2 \) and \( p < q \) the ratio of Gamma functions behaves as
\[ \frac{\Gamma \left( \frac{p+2}{\alpha} \right)}{p \Gamma \left( \frac{p}{\alpha} \right)} > \frac{\Gamma \left( \frac{q+2}{\alpha} \right)}{q \Gamma \left( \frac{q}{\alpha} \right)}. \] (2.63)

**Proof:** The proof goes along similar lines as that of Lemma 2.2. We note that for \( \alpha > 2 \) and from the definition given by Eq. (2.55), we have \(-1 < \delta < 0\). As a result of this, the first two factors on the RHS of Eq. (2.61) can be negative. However, they are either both negative or both positive, thus making the product positive. The sum in Eq. (2.62) is always negative for the \( \delta \) indicated, making \( f'(m) \) negative for all \( m > 0 \). \( \square \)

### 2.4 Properties of the Kurtosis

Beside being a key parameter in signal classification, the kurtosis of a source signal is an indicator of the separation difficulty in blind signal separation and deconvolution if moment-based criteria are employed. Whereas the maximum kurtosis is unbounded, the lower bound on the minimum kurtosis is \(-2\). Furthermore, the kurtosis of a composite signal has bounds that depend on the kurtoses of the constituent signals [96]. The influence of sampling-time accuracy to the kurtosis of the sampled signal has a direct implication on how accurate timing has to be in order for blind algorithms still to work satisfactorily.
2.4.1 Motivation

In blind signal processing, the kurtosis has two major functions. Due to the Central Limit Theorem, strong mixture and convolution push a composite distribution towards the normal distribution. recovering these signals means driving their distributions away from the normal distribution towards their original distribution, thereby separating or deconvolving the signals. This means increasing the kurtosis if the original source distribution is super-Gaussian, and decreasing the kurtosis for sub-Gaussian distributions. Hence, the kurtoses of the source signals are on the one hand a difficulty measure of the signals to separate, with a small absolute value indicating high difficulty, and the kurtoses of the separated signals, on the other hand, can serve as an objective function for the adaptation algorithm and as a convergence indicator. For the constant-modulus algorithm (CMA)—one of the most widely used blind deconvolution algorithm—the gradient of the CM cost surface is dependent on the source kurtosis [71]. Moreover, Johnson et al. [71] show the direct implication of the kurtosis to the convergence time of the CMA.

Zervas and Proakis [138] have pointed out the influence of constellation shaping on the kurtosis. However, by shaping the constellation in an effort to maximize capacity—the maximum capacity is reached for a Gaussian distribution—the signal becomes more and more difficult to separate. In this sense, the goals of fast convergence time and of high capacity are contradictory.

Finally, modulation classification is yet another use of the kurtosis [123]. For further comments on the use of the kurtosis, see also [94].

2.4.2 Minimum kurtosis

The fact that the lower limit of the kurtosis is $-2$ has been known for some time [81]. In this section we follow a slightly different route from that pursued in [81] to prove this result, which uses similar properties of crossing pdfs as a lemma proved later in the thesis. First we state a lemma on the fourth-order moment of a random variable, which was proved by Fimucan [53] in 1964.

**Lemma 2.3**

For any two symmetrical probability density functions $p_x(x)$ and $p_{\tilde{x}}(x)$ of a random variable $x$ with zero mean, variance $\sigma_x^2$, and the relations

$$p_x(x) \leq p_{\tilde{x}}(x), \quad \text{for} \quad a \leq |x| \leq b,$$

$$p_x(x) > p_{\tilde{x}}(x), \quad \text{otherwise},$$

the fourth-order moment $M_4 = E\{x^4\}$ for the distribution $p_x$ is greater than for $p_{\tilde{x}}$.

For an illustration of the relationship of the pdfs, see Fig. 2.5.

![Figure 2.5: Normalized Gaussian ($p_x$, solid line) and uniform ($p_{\tilde{x}}$, dashed line) distributions, illustrating Eq. (2.64).](image)

**Proof:** Since both functions $p_x(x)$ and $p_{\tilde{x}}(x)$ are pdfs and must therefore integrate to one, there exists a $c$ with $a \leq c \leq b$ such that

$$\int_0^c p_x(x) \, dx = \int_0^c p_{\tilde{x}}(x) \, dx.$$  

(2.65)
Likewise,
\[ \int_{-\infty}^{\infty} p_x(x) \, dx = \int_{-\infty}^{\infty} p_\mathcal{E}(x) \, dx. \]  
(2.66)

A second random variable \( y \) is now defined as
\[ y = (x^2 - c^2)^2. \]  
(2.67)

The mean of \( y \),
\[ E\{y\} = M_4 = 2c^2 \sigma_x^2 + c^4, \]  
(2.68)
is greater for the \( p_x \) than for the \( p_\mathcal{E} \) distribution, because of the higher probability density \( p_x \) in the range \( |x| < a \) of higher \( y \) values and an equal deficiency of probability in the range \( a < |x| < c \) of lower \( y \) values. The same argument is true for the ranges \( |x| > b \) and \( c < |x| < b \), respectively. A greater \( E\{y\} \) for \( p_x \) than for \( p_\mathcal{E} \) is equivalent to a greater fourth-order moment \( M_4 \), since \( c \) and \( \sigma_x \) are constants. □

With this result, we are now ready to prove a corollary on the minimum kurtosis, which is of interest to the study of digital communications signals.

**COROLLARY 2.3** (minimum kurtosis)

The kurtosis of a random variable cannot be less than \(-2\).

*Proof:* The value of \(-2\) for the kurtosis of a real random variable can be achieved by a 2-PAM (pulse amplitude modulation) signal, which is equivalent to a BPSK (binary phase shift keying) signal, given by its pdf
\[ p_\mathcal{E}(x) = \frac{1}{2} (\delta(x + 1) + \delta(x - 1)). \]  
(2.69)
The constants \( a, b, \) and \( c \) used in the proof of Lemma 2.3 are in this case equal, i.e., \( a = b = c = 1 \). The corollary follows immediately from the lemma above. Any distribution \( p_x \) different from \( p_\mathcal{E} \) (but with equal variance) has a higher probability density outside the region \( a < |x| < b \), which is concentrated to a single point. Therefore, its fourth-order moment and consequently its kurtosis are higher than that of the 2-PAM distribution \( p_\mathcal{E} \). □

An alternative proof was offered in [81] using the Cauchy-Schwarz inequality.

**2.4. Properties of the kurtosis**

*Proof (alternative):* We use the Cauchy-Schwarz inequality
\[ E\{A^2\} E\{B^2\} \geq (E\{AB\})^2, \]  
(2.70)
with equality if and only if \( A = cB, c \) a constant. With \( A = (x - m_x)^2 \) and \( B = 1 \), we get
\[ E\{(x - m_x)^4\} \geq \left( E\{(x - m_x)^2\} \right)^2. \]  
(2.71)
Dividing both sides of Eq. (2.71) by its right-hand side, we get
\[ \left( \frac{E\{(x - m_x)^4\}}{E\{(x - m_x)^2\}} \right)^2 = \frac{M_4}{\sigma_x^4} \geq 1, \]  
(2.72)
which is the same as
\[ \kappa_4 = \frac{M_4}{\sigma_x^4} - 3 \geq -2, \]  
(2.73)
with equality if and only if \( A \) is a constant. This, however, means that the pdf of \( x \) is concentrated at two values due to a further symmetry constraint. □

**2.4.3 Sum kurtosis**

We will now present a lemma on the sum of two independent random variables, which is related in nature to the Central Limit Theorem, and is of interest in connection with filtered signals.

**LEMMA 2.4**

The absolute value of the kurtosis of a sum of two independent zero-mean random variables with finite variances and kurtoses all \( \neq 0 \) is always smaller than the larger absolute value of the kurtoses of the two signals.

A proof of this lemma can be found in Appendix C.1.

An example shall illustrate this situation: Assume we have two independent discrete sources each having a 2-PAM distribution. The cumulants of the sum of independent signals are equal to the sum of their respective cumulants [106],...
so in our case the fourth-order cumulant doubles. However, since the variance, which is nothing but the second-order cumulant, also doubles, and the kurtosis is normalized by the squared variance, the kurtosis halves, hence in accordance with both the principle of minimum kurtosis stated in Corollary 2.3 and Lemma 2.4.

Shalvi and Weinstein [120] have made use of a related result to prove that if the kurtosis is equal to the source kurtosis at the output of an equalized system, then the original signal has been recovered.

Note that depending on $\sigma_1^2$ and $\sigma_2^2$, the respective variance for $x_1$ and $x_2$, either case

$$\min(|\kappa_4(x_1)|, |\kappa_4(x_2)|) < |\kappa_4(x_1 + x_2)| < \max(|\kappa_4(x_1)|, |\kappa_4(x_2)|)$$  \hspace{1cm} (2.74)

or

$$|\kappa_4(x_1 + x_2)| < \min(|\kappa_4(x_1)|, |\kappa_4(x_2)|) < \max(|\kappa_4(x_1)|, |\kappa_4(x_2)|)$$  \hspace{1cm} (2.75)

is possible. Table 2.5 shows the eight possible cases for the kurtosis of a sum of two independent signals. The stronger of the two signals in terms of power has a dominant influence on the kurtosis of the sum signal. For signals whose kurtoses have equal signs, we can develop a lower bound similar to the upper bound given by Lemma 2.4.

**Lemma 2.5**

The absolute value of the kurtosis of a sum of two independent zero-mean random variables with finite variances and equal-sign kurtoses both $\neq 0$ is always higher or equal than the lower absolute value of the kurtoses of the two signals divided by two.

A proof of this lemma can also be found in Appendix C.1.

### 2.4.4 Influence of timing offsets

Up to this point we have concentrated on ideally, critically sampled signals without a timing offset. In a realistic system we may find the situation where either clock recovery has not been carried out yet or shows some residual timing offset. This will affect the effective kurtosis and makes the signals harder to be separated.

<table>
<thead>
<tr>
<th>Examples</th>
<th>$\kappa_4(x_1 + x_2)$</th>
<th>$\kappa_4(x_1)$</th>
<th>$\kappa_4(x_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_1^2 &lt; \sigma_2^2$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_1)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
</tr>
<tr>
<td>$\sigma_1^2 &gt; \sigma_2^2$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_1)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
</tr>
<tr>
<td>$\sigma_1^2 \approx \sigma_2^2$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_1)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1 + x_2)$ &lt; $\kappa_4(x_2)$</td>
</tr>
<tr>
<td>$\kappa_4(x_1)$, $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1)$, $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1)$, $\kappa_4(x_2)$</td>
<td>$\kappa_4(x_1)$, $\kappa_4(x_2)$</td>
</tr>
</tbody>
</table>

Table 2.5: Kurtosis of a sum of two independent signals.
Communication signals are often pulse shaped by a raised-cosine filter with the impulse response

\[ h_k = \frac{\sin((k + \tau/T)\pi)}{(k + \tau/T)\pi} \cos(\rho(k + \tau/T)\pi) }{1 - 4\rho^2(k + \tau/T)^2}, \]  

with \( \rho \) being the roll-off factor, \( 0 \leq \rho \leq 1 \). The timing offset is denoted by \( \tau \), and \( T \) is the symbol duration. The impulse response for different \( \rho \) is shown in Fig. 2.6. If a random signal can be written as a weighted sum of independent random signals, the total \( n \)th-order cumulant is the weighted sum (with the original weights raised to the power of \( n \)) of all \( n \)th-order cumulants of the individual signals\(^3\). For an i.i.d. process, we can therefore calculate the kurtosis of the filter output as

\[ \kappa_{4,\text{out}} = \frac{\sum_k h_k^4 }{(\sum_k h_k)^2} \cdot \kappa_{4,\text{in}}, \]  

(2.77)

\(^3\)This property, among others, is the reason why cumulants are sometimes referred to as semiinvariants.

Figure 2.6: Impulse response of the raised-cosine filter for different roll-off factors \( \rho \).

where \( \kappa_{4,\text{in}} \) denotes the kurtosis of the input signal to the filter. For all possible values of the timing offset \( \tau \), the weighting factors \( h_k \) are maximal for \( \rho = 0 \) (ideal lowpass filter). For \( \rho > 0 \), the intersymbol-interference terms are smaller as a consequence of the fact that

\[ \frac{\cos(\rho(\tau/T)\pi)}{1 - 4\rho^2(\tau/T)^2} \geq \frac{\cos(\rho(k + \tau/T)\pi)}{1 - 4\rho^2(k + \tau/T)^2}, \quad 0 \leq \frac{\tau}{T} \leq 0.5, \quad k = \{\pm1, \pm2, \ldots\}. \]  

(2.78)

which shall not be proved here. For this filter (or any of the form given by Eq. (2.76)), the largest degradation of the kurtosis occurs for a timing offset of half a symbol \( (\tau = 0.5T) \). For \( \tau = 0.57T \), the worst-case kurtosis of the filtered signal is then given by

\[ \kappa_{4,\text{out}} = \frac{\sum_k \sin^4((k + 0.5)\pi\tau)}{(k + 0.5)^4\pi^4} \cdot \kappa_{4,\text{in}}. \]  

(2.79)

Using the equalities [21]

\[ \sin((k + \tau/T)\pi) = (-1)^k \sin(\pi\tau/T), \]  

(2.80)

\[ \sum_{k=0}^{\infty} \frac{1}{(2k+1)^2} = \frac{\pi^2}{8}, \]  

(2.81)

and

\[ \sum_{k=0}^{\infty} \frac{1}{(2k+1)^4} = \frac{\pi^4}{96}, \]  

(2.82)

in Eq. (2.79), we obtain

\[ \kappa_{4,\text{out}} = \frac{\kappa_{4,\text{in}}}{3} \cdot \rho = 0, \quad \tau/T = 0.5. \]  

(2.83)

In fact, it can be shown that for \( \rho = 0 \), the evaluation of Eq. (2.77) yields

\[ \kappa_{4,\text{out}} = \frac{1 + 2\cos^2(\pi\tau/T)}{3} \cdot \kappa_{4,\text{in}} \]  

\[ = \frac{2 + \cos(2\pi\tau/T)}{3} \cdot \kappa_{4,\text{in}}. \]  

(2.84)
The smallest degradation of the kurtosis occurs for $\rho = 1$. For this roll-off factor and the worst-case degradation ($\tau / T = 0.5$), $h_k = 0$ for all $k$ but for $k = 0$ and $k = -1$. For $h_0$ we get by applying Bernoulli-de l’Hospital’s rule

$$h_0 = \lim_{\tau / T \to 0.5} \left( \frac{\sin \frac{\pi}{2} \cos \left( \frac{\pi}{2} \tau \right)}{1 - 4 \left( \frac{\tau}{T} \right)^2} \right)$$

$$= \frac{2}{\pi} \left( -\tau \sin \left( \frac{\pi}{2} \tau \right) - \frac{8 \tau}{T} \right)_{\tau / T = 0.5} = 0.5. \quad (2.85)$$

Similarly, it can be found that $h_{-1} = h_0 = 0.5$. Thus, we get for the kurtosis transformation after Eq. (2.77)

$$\kappa_{4,\text{out}} = \frac{\kappa_{4,\text{in}}}{2}, \quad \rho = 1, \quad \tau / T = 0.5. \quad (2.86)$$

Fig. 2.7 shows the degradation of the kurtosis caused by timing misadjustment for the raised-cosine filter with different roll-off factors $\rho$. Note that this degradation is only relative to the kurtosis of the distribution, and it is furthermore independent of the distribution. The ideal lowpass filter can clearly be identified as the degradation bound (Eq. (2.84)) on the kurtosis, depicted by the solid line. For sub-Gaussian signals, Eq. (2.83) means that no matter what timing offset we have (or even if clock recovery has not yet taken place), the kurtosis is upper bounded by $\kappa_4 / 3$. Blind adaptive methods may therefore still be successfully applied. Valkama et al. [129] show the influence of timing offsets on the performance of a blind separation algorithm.

### 2.4.5 SNR estimation

**Signal-to-noise ratio estimation** is a task frequently used in communication algorithms. In antenna diversity, if maximum-ratio combining is applied, the combination weights depend on the SNR of the individual branches. SNR also serves as a cell hand-over criterion in a cellular network. Usually, the signal strength at the receiver input serves as an indicator of the SNR, which is often justified if the latter is dominated by thermal noise of the receiver. Many other noise sources or interferers are possible, and other more meaningful measures to estimate the SNR are needed. If the noise is of Gaussian nature, and the timing is accurate, but demodulation has not yet been fully carried out, an SNR estimator can be derived using the kurtosis. To this end we observe the kurtosis of the sum of two independent signals (see Appendix C.1)

$$\kappa_4(x_1 + x_2) = \frac{\kappa_4(x_1) \sigma_1^4 + \kappa_4(x_2) \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2}. \quad (2.87)$$

If we consider $x_1$ to be the signal of interest and $x_2$ additive white Gaussian noise, then by using $\kappa_4(x_2) = 0$ and $\sigma_2^2 = \sigma_1^2 + \sigma_2^2$, we get

$$\sigma_1^2 = \sigma_2^2 \sqrt{\left| \frac{\kappa_4(x_1 + x_2)}{\kappa_4(x_1)} \right|}, \quad (2.88)$$

from which we can arrive at the signal-to-noise ratio

$$\text{SNR} = \frac{\sigma_1^2}{\sigma_2^2} = \sqrt{\frac{\kappa_4(x_1 + x_2)}{\kappa_4(x_1)}} \frac{1}{\sqrt{\kappa_4(x_1) - \kappa_4(x_1 + x_2)}}. \quad (2.89)$$

If the expression for the kurtosis of the received signal $r = x_1 + x_2$ is replaced

![Figure 2.7: Degradation of the kurtosis caused by timing misadjustment for a raised-cosine pulse-shaping filter.](attachment:figure2_7.png)
by the sample kurtosis, we get the SNR estimate

$$\text{SNR} = \frac{\sqrt{\frac{1}{N} \sum r_i^4 - 3 \left( \frac{1}{N} \sum r_i^2 \right)^2}}{\sqrt{|\kappa_4(x_1)|} \left( \frac{1}{N} \sum r_i^2 \right)^2}, \quad (2.90)$$

where \(r_i\) denotes the \(i^{th}\) sample of the received signal. Eq. (2.90) forms an estimator of the SNR of a signal subject to additive Gaussian noise.

### 2.5 Summary

Parametric distributions help model real-world signals and investigate suitability of blind algorithms, since important fourth-order statistics (kurtosis) are directly controlled by the parameter of the distribution.

From all possible continuous and discrete distributions, \(M\)-PSK signals have the lowest possible kurtosis \(\kappa_4 = -2\), are therefore farthest from Gaussian, and hence build the most suitable class of digitally modulated signals for blind algorithms using higher-order statistics. The kurtoses of other modulation schemes can be expressed in a parametric closed form, and are all between \(-2\) and \(-0.6\). If timing offsets exist, the kurtosis might be maximally degraded to \(1/3\) of its original value for a certain class of pulse-shaping filters. Other bounds for composite signals are intuitive regarding the Central Limit Theorem.

Apart from a convergence indicator and a measure of difficulty in blind techniques, the kurtosis may be used in estimation theory. As an example, the SNR can be estimated for an additive white Gaussian noise channel using the sample kurtosis.

### Chapter 3

**Blind signal separation**

In a general scenario with channel noise, the blind signal separation problem, where the estimated signals are extracted from mixtures of the original signals, can only be solved in a probabilistic sense, due to the lack of knowledge of the mixing conditions. This means that we can at best answer the question as to what mixing conditions most likely produced the current observation. The answer might not be unique, but more importantly, the question often is computationally prohibitive to answer. To overcome this inaccessibility of maximum-likelihood (ML) related solutions, adaptive algorithms using stochastic gradients are employed.

After the problem formulation (Section 3.1), such an adaptive stochastic-gradient algorithm based on the ML rule will be introduced in Section 3.2. The natural-gradient approach has found wide acceptance as a fast variant of stochastic-gradient algorithms, and will be investigated in Section 3.3. Section 3.4 and Section 3.5 deal with the convergence behavior and equilibrium points of the separating solution, respectively. Other criteria than ML, collected in Section 3.6, lead to similar solutions, whose validity can be checked using local stability analysis, as derived in Section 3.7. In this section, stability regions of some widely used nonlinearities are also provided. Finally, overdetermined signal separation and other modifications will be treated in Section 3.8 and Section 3.9, respectively.
3.1 Problem formulation

3.1.1 Model and assumptions

The problem of blindly separating an instantaneous mixture of signals can be described as follows (see also Fig. 3.1). Suppose that the information sources generate a number of signals, conveniently described by the vector $s = [s_1, \ldots, s_{M_s}]^T$. Through a mixing process, usually presumed linear, and therefore represented by an unknown scalar matrix $A$, and an additive noise vector $n = [n_1, \ldots, n_{M_s}]^T$, we get the observation vector $x = [x_1, \ldots, x_{M_s}]^T$ at the sensors. $M_s$ denotes the number of sources as well as the number of sensors.

Mathematically, we describe the observed signals by

$$x = As + n. \quad (3.1)$$

We attack the problem by attempting to find a scalar matrix $W$, describing the separation process, such that the signals in vector $u = [u_1, \ldots, u_{M_s}]^T$ are noisy replicas of the original source signals up to some ambiguities. These ambiguities are:

- the order of the signals within vector $u$ (permutation)
- the magnitudes of the original source signals (scaling)
- the phases of the original source signals (the signs for real-valued sources)

Figure 3.1: Adaptive blind signal separation system. All data paths are vectors.

As the noise, sometimes knowledge of $\sigma_n^2$ is assumed. Often, as we will see, the noise is considered negligible, thus $\sigma_n^2 = 0$ is assumed. In this case, the model of linear mixture and separation is often the one given in Fig. 3.2.

In the more general case of considering channel noise, the recovered signals can be written as

$$u = Wx = W(As + n) = Ps + Wn. \quad (3.2)$$

If the inverse of $A$ is desired, $P = WA$ should approximate as closely as possible a scaled permutation matrix. Since the numbering of the sources and for
symmetric distributions also the sign cannot be recovered, the number of possible permutation matrices grows rapidly with the number of sources, \( M_s \). In fact, this number is

\[
\#P = M_s! \cdot 2^{M_s}. \quad (3.3)
\]

For illustration purposes of the large degree of freedom the number of possible permutation matrices is given in Table 3.1.

<table>
<thead>
<tr>
<th>( M_s )</th>
<th>( #P )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>48</td>
</tr>
<tr>
<td>4</td>
<td>384</td>
</tr>
<tr>
<td>5</td>
<td>3840</td>
</tr>
<tr>
<td>6</td>
<td>46080</td>
</tr>
<tr>
<td>( M_s \gg 1 )</td>
<td>( \sqrt{2 \pi M_s \left( \frac{2M_s}{e} \right)^{M_s}} )</td>
</tr>
</tbody>
</table>

Table 3.1: Number of possible permutation matrices as a function of the number of sources. The last entry is an approximation using Stirling’s formula [21].

The separation process can be modeled as a single-layer neural network (see for example [4] or [102]) with an equal number of input and output nodes, where the coefficients \( w_{ij} \) of the separation matrix \( W \) are simply the weights from the input to the output nodes. The activation functions—the equivalent term for nonlinear function or nonlinearity in the neural network terminology—at the output nodes are used for the training mode only. For the separation process itself we restrict ourselves to the linear case. This is particularly important for acoustic applications, where nonlinear signal processing might generate unacceptable audible distortion. In the case of substantial noise, a nonlinear transformation might yield better results (as far as MSE criteria are concerned).

Throughout the thesis we will restrict ourselves to cases where inherent higher-order statistics (HOS) produced by nonlinearities are unavoidable since second-order statistics (SOS) are insufficient, i.e., we assume critically sampled, white signals.

### 3.1.2 Performance measures

In order to find a meaningful fidelity measure to judge the separation quality, a scalar number is needed that describes the average degree of the residual mixing of the separated source signals at the output of the network. Amari et al. [5] define the performance index as a ratio of the sum of absolute values of contributions of the unwanted source signals leaking through and the absolute value of the desired source. In slight modification to this, the performance measure may be defined based on the power of signals rather than on absolute values [73]. Ideally, the scaled permutation matrix \( P = [p_{ik}] \) in Eq (3.2) would have one nonzero entry in each row as well as one nonzero entry in each column. The value \( |p_{ik}|^2 \) is the power transfer from source \( s_k \) to the output \( u_i \). Realistically, when separation is achieved, in each row there will be one dominant entry (\( \max |p_{ik}|^2 \)), indicating the power transfer of the separated source \( s_k \). The other entries of the same row of \( P \), which end up being nonzero in a realistic case, if squared and summed, reveal the power of other source signals leaking through to the particular output if all source signals have equal power. Vice versa, by squaring the subdominant entries of the same column of \( P \), we get the power of one source leaking through to different outputs. Provided the same-source lock-on is not a problem for the algorithm under consideration [85], the row-wise observation of the permutation matrix is more meaningful than the column-wise observation, rendering the performance index into a measure of interchannel interference (ICI) [73]

\[
J_{\text{ICl}}(P) \triangleq \frac{1}{M_s} \sum_{k=1}^{M_s} \sum_{i=1}^{M_s} \frac{|p_{ik}|^2 - \max_k |p_{ik}|^2}{\max_k |p_{ik}|^2} = \frac{1}{M_s} \left( \sum_{i=1}^{M_s} \sum_{k=1}^{M_s} |p_{ik}|^2 \right) - 1. \quad (3.4)
\]

Of course, \( J_{\text{ICl}}(P) \) is available in a simulation environment only. In practical situations, the true matrix \( A \) and therefore, the matrix \( P \) is unknown.

### 3.1.3 Narrowband model

In the previous section we assumed an instantaneous mixture model, i.e., the mixing process is fully described by attenuation scalars. In reality, we often have additional individual delays for the different propagation paths. If delay
differences of the different paths are large, our model is no longer valid. This happens frequently in acoustic applications, when the sensor displacement is on the order of the wavelength or the maximum wavelength difference of the frequency band in use. In this case, more complicated models must be sought employing delays [125] or even FIR filters [84].

On the other hand, there are many applications, e.g., in wireless communications, where, despite the much larger sensor displacement compared to the wavelength, the signal bandwidth of interest is small enough for all frequencies to incur the same phase differences. This can be seen as follows. In addition to each element $a_{ik}$ of the matrix $A$ representing the attenuation of a particular path, we define a time delay $\tau_{ik}$, collected in the matrix $T$. The output of the channel can now be described as

$$x = (A \circ \exp(-j \omega_{RF}T))x + n.$$

where $\omega_{RF}$ is the frequency of operation and $\circ$ denotes the element-wise multiplication of two matrices. Since we have an element-wise operation, we can now combine amplitude and phase information into the matrix $A$ by writing its $(i,k)$th element as $a_{ik} e^{-j\phi_{ik}}$, with $\phi_{ik} \equiv \omega_{RF} \tau_{ik}$ being the phase associated with the $(i,k)$th path. The matrix $A$ is now a matrix of complex scalars.

### 3.1.4 Relationship to beamforming

Although beamforming antenna arrangements and the physical sensor setup of blind signal separation algorithms usually differ considerably—the antenna spacing of the former technique is normally restricted by Nyquist spatial sampling considerations, whereas for the latter no such restrictions apply—there are common aspects of both techniques. The blind signal separation methods are one of many methods to steer beams toward the signal of interest. This is referred to as blind estimation of co-channel signals [59].

### 3.2 Maximum-likelihood solution

One possible solution to the problem stated in Section 3.1 can be found by answering the question of what mixing matrix has most likely led to the current observation $x$. Our goal is to find the inverse of the mixing matrix, $W = A^{-1}$. This is a zero-forcing solution, since it nulls any contribution from sources other than the one of interest. An MMSE solution, which may make more sense in a noisy environment, will be directly computable from this solution of $W$, as will be shown later. We will now derive the maximum-likelihood solution, which in turn gives us the desired separation matrix $W$.

We consider the likelihood as the probability density of the observation, parameterized by $A$, hence the conditional probability density $p_{x|A}(x|A)$. If the noise signals were known, we could write the conditional probability

$$p_{x|A,n}(x|A,n) = p_x(A^{-1}(x-n))$$

The noise vector $n$ is a nuisance parameter, which we do not know. To get the unconditional probability we have to get rid of the nuisance parameter by integrating over it

$$p_x(x|A) = \int_{-\infty}^{\infty} p_{x|A,n}(x|A,n) p_n(n) dn$$

Owing to the mutual-independence assumption of the source signals, we can factorize the pdf of the source signals

$$p_s(s) = \prod_{i=1}^{M_s} p_s(s_i),$$

with $p_s(.)$ denoting the pdf of the $i$th source. Likewise, for the noise vector we have

$$p_n(n) = \prod_{k=1}^{M_n} p_n(n_k),$$

with $p_n(.)$ being the pdf of the noise source at sensor $k$. The log-likelihood function $L$ is then given by

$$L \triangleq \ln p_x(x|A) = \ln |\det W| + \ln \int_{-\infty}^{\infty} p_s(u - Wn) p_n(n) dn$$

$$= \ln |\det W| + \ln \int_{-\infty}^{\infty} \prod_{j=1}^{M_s} p_s(u_j - W n_j) \prod_{k=1}^{M_n} p_n(n_k) dn_1 \ldots dn_{M_n}.$$ (3.10)
where $w_i^T$ is the $i$th row of the matrix $W$. As with most ML-related solutions, Eq. (3.10) is difficult to solve directly. An adaptive solution using a gradient-search method is usually sought to overcome this problem. In order to find a gradient leading to the ML solution, we have to differentiate $L$ w.r.t. the matrix $W$. We write this gradient element-wise

$$
\frac{\partial L}{\partial w_{mn}} = [W^{-T}]_{mn}
$$

$$
\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_m'(u_m - w_m^T n) \cdot (x_n - n) \prod_{i=1}^{M_1} p_i(u_i - w_i^T n) \prod_{k=1}^{M_1} p_k(n_k) \, dn_1 \ldots dn_{M_2}
$$

$$
+ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \prod_{i=1}^{M_1} p_i'(u_i - w_i^T n) \prod_{k=1}^{M_1} p_k(n_k) \, dn_1 \ldots dn_{M_2}
$$

(3.11)

where $[W^{-T}]_{mn}$ is the $(m, n)$th entry in $W^{-T}$. As an illustration of the form of Eq. (3.11), the gradient of $w_{12}$ is given for the case of $M_2 = 2$ sources

$$
\frac{\partial L}{\partial w_{12}} = \frac{-w_{12}}{w_{11} w_{22} - w_{12} w_{21}} + \frac{-w_{12}}{w_{11} w_{22} - w_{12} w_{21}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_1'(u_1 - w_1 n_1 - w_2 n_2) \cdot (x_2 - n_2) p_2'(u_2 - w_1 n_1 - w_2 n_2) p_2(n_1) p_2(n_2) \, dn_1 \, dn_2
$$

$$
+ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_1'(u_1 - w_1 n_1 - w_2 n_2) p_2'(u_2 - w_1 n_1 - w_2 n_2) p_1(n_1) p_2(n_2) \, dn_1 \, dn_2
$$

(3.12)

In practice, Eq. (3.11) is too complicated for a real-time implementation. If the noise is negligible, however, Eq. (3.11) turns into a much easier form

$$
\frac{\partial L}{\partial w_{mn}} = [W^{-T}]_{mn} + \frac{p_m'(u_m)}{p_m(u_m)} \cdot x_n
$$

(3.13)

$$
= [W^{-T}]_{mn} - g_m(u_m) x_n
$$

(3.14)

where

$$
g_m(u_m) = -\frac{\partial \ln p_m(u_m)}{\partial u_m} = -\frac{p_m'(u_m)}{p_m(u_m)}, \quad i = 1, \ldots, M_s
$$

(3.15)

with $p_m(u_m)$ and $p_m'(u_m)$ being the source pdf and its derivative, respectively. The function $g_m()$ is called the score function associated with a certain pdf. A possible update equation for the separation matrix using a stochastic gradient can now be formulated from Eq. (3.14) as

$$
W_{t+1} = W_t + \mu \left( W_t^{-T} - g(u)x^T \right)
$$

(3.16)

where $\mu$ is a small step size, and $g(u) = [g_1(u_1), \ldots, g_{M_s}(u_{M_s})]^T$ is the vector of score functions. Similar ML approaches, also without taking noise into account, have been suggested by Pham and Garat [112]. The direct derivation of Eq. (3.16) in the noiseless case was also given by Yang [137].

As examples of the score functions, we give in Table 3.2 the score functions of the parametric distribution families introduced in Chapter 2.

<table>
<thead>
<tr>
<th>distribution family</th>
<th>score function</th>
</tr>
</thead>
<tbody>
<tr>
<td>generalized Gaussian</td>
<td>$g_i(u_i) = \frac{\alpha}{\beta \eta}</td>
</tr>
<tr>
<td>two-tailed Gamma</td>
<td>$g_i(u_i) = \left( \sqrt{\alpha + 1} \alpha - \frac{\alpha - 1}{</td>
</tr>
<tr>
<td>unbiased symmetric Beta</td>
<td>$g_i(u_i) = \frac{2(\alpha - 1)u_i}{2\alpha + 1 - u_i^2}$</td>
</tr>
</tbody>
</table>

Table 3.2: Score functions of some parametric distribution families.

### 3.3 Accelerated update equation

#### 3.3.1 Natural/relative gradient

The convergence of Eq. (3.16) is not very fast and depends on the mixing matrix $\mathbf{A}$ as well as on the initial $W_{t=0}$. Besides, the implementation of Eq. (3.16) involves a matrix inversion, an operation that should be avoided for fast real-time algorithms. Possible ways out of these problems were presented by Amari [2] and Cardoso [22] by using the natural gradient and the relative gradient, respectively. The natural gradient corrects for the nonisotropic gradient magnitude structure—called Riemannian structure in information geometry—in the

Note that we mean $x = x_i$ and $u = u_i$ here and for the rest of the thesis unless indicated otherwise but omit the index $i$ for better legibility except for the separation matrix $W$, where it emphasizes the iteration process.
parameter space of the standard-gradient adaptation, but at the same time preserves local minima of the cost function. For the blind separation problem, the natural-gradient (as well as the relative-gradient) method involves a postmultiplication of the matrix update by $W^T W$, hence Eq. (3.16) becomes

$$ W_{t+1} = W_t + \mu \left( I - g(u)u^T \right) W_t, $$

(3.17)

thereby getting rid of the matrix inversion. Moreover, the convergence speed of Eq. (3.17) is considerably improved over the original update equation, i.e., Eq. (3.16). We will refer to Eq. (3.17) as the asymmetric ICA [24] throughout this thesis. A comparison between Eq. (3.16) and Eq. (3.17) and further details on the natural gradient and its properties are given in [45].

3.3.2 Alternative derivation of the gradient modification

The update equation, Eq. (3.17), can also be obtained via a different path [73] by exchanging a nonblind criterion of an RLS-like algorithm by a blind criterion so that we get an update equation of the following form

$$ W_{t+1} = W_t + \mu \left( I - g(u)u^T \right)^{-1} \left( I - \mu \left( I - g(u)u^T \right) g(u)u^T \right) W_t. $$

(3.18)

By applying the Matrix-Inversion Lemma (see Appendix A.3) to Eq. (3.18), we get

$$ W_{t+1} = W_t + \mu \left( I - \frac{1}{1 - \mu + \mu u^T g(u) g(u)u^T} \right) W_t. $$

(3.19)

Alternatively, under the assumption that $\mu \left( I - g(u)u^T \right)$ has a spectral norm smaller than one, which is usually achieved by choosing $\mu$ small enough, we can expand Eq. (3.18) into an infinite sum

$$ W_{t+1} = W_t + \sum_{k=1}^{\infty} \left( \mu \left( I - g(u)u^T \right) \right)^k W_t. $$

(3.20)

With small step sizes $\mu$, higher terms in Eq. (3.20) decrease rapidly. Neglecting terms with $\mu^k$ and higher orders of $\mu$, Eq. (3.20) goes over into Eq. (3.17).

Going back to the original asymmetric ICA, there are three key issues that need further discussion:

- natural/relative gradient (convergence)
- equilibrium point (independence)
- local stability

The first point has a great influence on the convergence speed as mentioned in this section. The second point involves the question whether the equilibrium points, for which

$$ E \left[ I - g(u)u^T \right] = 0, $$

(3.21)

imply a location within the separation-matrix manifold where the output signals are mutually independent, and will be addressed in the next section. The last point, finally, answers the question whether equilibrium points are stable in the sense that small deviations of the separation matrix from such a point will pull it back to the separating solution. Global stability is difficult to investigate due to a complicated cost structure in the parameter manifold. The next best thing is local stability. In Section 3.7, sufficient but not necessary conditions for local stability of the natural-gradient solution, Eq. (3.17), are provided. The advantage of these conditions is that they are decoupled with respect to the individual source distributions. The actual conditions are weaker but involve equations with pairwise appearance of the source distributions.

3.4 Convergence behavior

Some computer simulations will illustrate the convergence behavior of the algorithms given above. All source signals in this simulation were independent Laplacian-distributed random signals. The hyperbolic tangent was used as the nonlinearity $g(\cdot)$, which is a common choice for super-Gaussian sources and is close enough to the true score function of a Laplacian distributed random variable. Explanations as to the proper choice of the nonlinearity are relegated to the next chapter. To generate a general, representative case, the matrix $A$ is chosen such that its singular values $\lambda_i$ are logarithmically distributed and its condition number is $\lambda_{\text{max}}/\lambda_{\text{min}} = 100$. The initial value for the separation matrix is chosen as $W_0 = I$. The desired final separation performance $J_{\text{ref}}(P)$, the residual mixing after convergence, determines the step size $\mu$ and has been found by simulations. In Fig. 3.3a) the dependency of the separation performance on the step size for ten unknown sources can clearly be observed. The
curves are averages over 50 typical runs with different input signals and mixing matrices. Fig. 3.3b) shows the singular values of $P$ for $J_{oo}(P) = -20$ dB, which reflects the progress of the decorrelation performance of the permutation matrix $P$. Because of

$$P_0 = W_0 \cdot A = A,$$  \hspace{1cm} (3.22)

where the subscript zero denotes the initial value of the respective matrix, at the beginning of the plot we clearly see the singular-value spread of $A$, hence of $P_0$. Very quickly the singular values converge to one, at which point the output signals are decorrelated (the condition number of the global transfer matrix at this point is $\chi(P) = 1$). However, at this stage the signals are still far from being separated as can be seen by comparing a) and b) of Fig. 3.3. It is due to this behavior that in the blind case the natural-gradient algorithm, Eq. (3.17), shows about the same convergence time as the algorithm according to Eq. (3.19), because the advantage of the latter only holds as long as the output signals are still correlated, as experiments indicated.

However, for moderate $J_{oo}$, e.g., $J_{oo}(P) = -10$ dB, Eq. (3.19) shows some advantage over Eq. (3.17) in terms of fewer outliers and a lower variance of $J_{oo}$ for the same average $J_{oo}$. A convergence plot with one typical realization of the settings as described above, is depicted in Fig. 3.4. The increased complexity of Eq. (3.19), which acts like some step-size control (or normalization), is worthwhile in this case.

![Figure 3.3](image1.png)

**Figure 3.3:** Performance of blind separation algorithm for ten Laplacian sources. a) Performance measure $J(P_b)$ as described by Eq. (3.4). b) Singular values of the permutation matrix $P$ for $J_{oo}(P) = -20$ dB.

![Figure 3.4](image2.png)

**Figure 3.4:** Convergence comparison of BSS algorithms. Upper plot: algorithm according to Eq. (3.17). Lower plot: algorithm according to Eq. (3.19).

### 3.5 Equilibrium points

Reformulating Eq. (3.21) means

$$E \{ g_i(u_i)u_k \} = \begin{cases} 1, & i = k, \\ 0, & i \neq k, \end{cases}$$  \hspace{1cm} (3.23)
expressing that $u_i$ and $u_k$ are independent unless $i = k$. If $g_i(\cdot)$ was a linear operator, thus transforming the update equation according to Eq. (3.17) into

$$W_{t+1} = W_t + \mu \left( I - uu^T \right) W_t,$$

(3.24)

we would only decorrelate the signals in $u$ (cf. [46]), as

$$E\{u_iu_k\} = \begin{cases} 1, & i = k, \\ 0, & i \neq k. \end{cases}$$

(3.25)

Decorrelation is a necessary but not sufficient condition for true independence. Under certain circumstances, such as nonstationary, cyclostationary, or temporally correlated source signals, second-order statistics such as those implicitly produced by Eq. (3.24) are sufficient. In general, however, this is not the case. For true independence, the joint distribution of two signals must be factorial

$$p_{u_iu_k}(u_i, u_k) = p_{u_i}(u_i) \cdot p_{u_k}(u_k).$$

(3.26)

From this it can easily be seen, that in this case the cross-moments are

$$E\{u_i^m u_k\} = 0, \quad i \neq k, \quad m \text{ odd},$$

(3.27)

because

$$E\{u_i^m u_k\} = \int_{-\infty}^{\infty} p_{u_iu_k}(u_i, u_k) u_i^m u_k \, du_i \, du_k
= \int_{-\infty}^{\infty} p_{u_i}(u_i) u_i^m \, du_i \cdot \int_{-\infty}^{\infty} p_{u_k}(u_k) u_k \, du_k
= E\{u_i^m\} E\{u_k\} = 0,$$

(3.28)

where the last equality is from the zero-mean assumption. If $g_i(\cdot)$ is a general nonlinearity, it can be expanded into its Taylor series, which produces higher-order cross-moments. We can then see from Eq. (3.28) that if the signals are independent to each other, all these cross-moments disappear, yielding Eq. (3.21).

3.6 Other cost functions

Many approaches have resulted in similar update equations for the separation matrix. Very frequently, the nonlinear functions $g_i(u_i)$ derived by different criteria are similar in nature for a given probability distribution of the source signals to separate. In the blind signal processing literature, criteria are usually called contrast functions or objective functions. The following criteria have been proposed for use in adaptive blind signal separation:

- maximum likelihood
- information maximization
- mutual information
- negentropy
- multiuser CMA
- kurtosis

Some of these criteria, although apparently different, lead to identical nonlinearities. A collection of cost functions for blind separation has been given by Joho et al. [72]. For a good overview of different criteria and their equivalence, see also [24] and references therein.

3.6.1 Entropy maximization / information maximization

In the earlier days of blind signal processing, the nonlinear functions used for blind signal separation were often chosen based upon heuristic considerations rather than profound theory. One of the first to suggest an information-theoretic criterion were Bell and Sejnowski [11, 12]. The entropy maximization\(^4\) [11] is based on the fact that for an output nonlinearity that transforms a signal distribution to the uniform distribution—the entropy-maximizing distribution under the assumption of finite amplitude—the maximization of the joint output entropy is reached if the mutual information between the outputs is minimized. Cardoso [23] pointed out the equivalence of the maximum likelihood (ML) and information maximization (Infomax) method. Formally, the joint output entropy is given by

$$H(y) \triangleq \sum_{i=1}^{M} H(y_i) - I(y_1; \ldots; y_M),$$

(3.29)

where

$$y_i \triangleq f_i(u_i)$$

(3.30)

\(^4\)The information-theoretic expressions used in this chapter and subsequently may be found in any book on information theory, e.g., [37].
is the output after the nonlinear function $f_i(.)$ that transforms $u_i$ into the uniform distribution. The nonlinear function that accomplishes this requirement is the cumulative distribution (with bias removed). Hence,

$$f_i(u_i) = \int_{-\infty}^{u_i} p_{s_i}(s_i) ds_i - 0.5. \quad (3.31)$$

The joint output entropy is

$$H(y) = -E\{\ln p_y(y)\} = -E\{\ln p_x(x)\} + E\{\ln |J|\}, \quad (3.32)$$

where the Jacobian $J$ is given by

$$J = \det W \cdot \prod_{i=1}^{M_s} f_i'(x_i). \quad (3.33)$$

The first term on the RHS of Eq. (3.32) is independent of $W$, hence

$$\frac{\partial}{\partial W} W H(y) = -E\{\ln |J|\} = (W^T)^{-1} + \frac{\partial}{\partial W} E\left\{\sum_{i=1}^{M_s} \ln f_i'(x_i)\right\}$$

$$= (W^T)^{-1} + E\left\{\sum_{i=1}^{M_s} f_i''(x_i) \frac{\partial u_i}{\partial W}\right\}. \quad (3.34)$$

By noting that $f_i'(.) = p_{s_i}(.)$ and $f_i''(.) = p_{s_i}''(.)$, respectively, we see that Eq. (3.34) leads indeed to the same update rule as that given by Eq. (3.16). Moreover, Bell and Sejnowski pointed out that this update equation consists of two nicely balancing terms. The first term $W^{-T}$ is an anti-decay term, which prevents the coefficients from vanishing. The second term is an anti-Hebbian term\(^5\), which punishes any higher-order cross-moments present in the outputs.

In a related publication, Bell and Sejnowski [12] showed an equivalent approach by maximizing the mutual information between the input and the output of a neural network, hence called information maximization (Infomax). This

\(^5\)A term is called Hebbian (+) or anti-Hebbian (−), respectively [69], if the update term is of the form $\Delta W \propto \pm g(W x) x^T + \ldots$.

---

### 3.6.2 Negentropy

The negentropy [57] is defined as the Kullback-Leibler divergence between a multivariate pdf $p_u(x)$ and its Gaussian counterpart $p_n$ with the same means and covariance matrix

$$D(p_u(x)\parallel p_n(x)) = \int_{-\infty}^{\infty} p_u(x) \ln \frac{p_u(x)}{p_n(x)} dx$$

$$= \int_{-\infty}^{\infty} p_u(x) \ln p_n(x) dx - \int_{-\infty}^{\infty} p_u(x) \ln p_n(x) dx. \quad (3.37)$$

Since $\ln p_n$ is a quadratic form, and up to second-order statistics $p_u$ and $p_n$ are identical, we have

$$\int_{-\infty}^{\infty} p_u(x) \ln p_n(x) dx = \int_{-\infty}^{\infty} p_n(x) \ln p_n(x) dx = -H(n). \quad (3.38)$$

Hence,

$$D(p_0(x)\parallel p_n(x)) = -H(u) + H(n), \quad (3.39)$$

or in words, the negentropy is given by the differences of the output entropy and the entropy of a Gaussian second-order equivalent. The application of the negentropy criterion to the BSS problems corresponds to the converse of the Central Limit Theorem, which states that a mixture of any signals will be closer to a Gaussian distribution than the original source signals. By maximizing the negentropy, we can therefore reach a state of separated signals. Girolami and Fyfe [57] showed that the stochastic-gradient update rule minimizing Eq. (3.39) leads again to Eq. (3.16).
3.6.3 Mutual information

As opposed to maximizing the output entropy, the approach by Amari et al. [5] minimizes the average mutual information (MI) between two signals at the output, an idea raised by Comon [36]. This measure is expressed by the Kullback-Leibler divergence between the joint output distribution and the product of the marginal output distributions. For two signals we can easily show the relationship by the following line of argument. Suppose that \( \mathbf{u} = [u_1, u_2]^T \) is a random vector of two output signals. Now the Kullback-Leibler divergence can be written as

\[
D(p_{u_1, u_2} \| p_{u_1} p_{u_2}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_{u_1, u_2}(u_1, u_2) \ln \frac{p_{u_1, u_2}(u_1, u_2)}{p_{u_1}(u_1) p_{u_2}(u_2)} \, du_1 \, du_2
\]

\[
= -H(u) - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_{u_1, u_2}(u_1, u_2) \ln \left( \frac{p_{u_1}(u_1)p_{u_2}(u_2)}{p_{u_1}(u_1) p_{u_2}(u_2)} \right) \, du_1 \, du_2
\]

\[
= -H(u) - \int_{-\infty}^{\infty} p_{u_1}(u_1) \ln p_{u_1}(u_1) \, du_1 - \int_{-\infty}^{\infty} p_{u_2}(u_2) \ln p_{u_2}(u_2) \, du_2
\]

\[
= -H(u_1) + H(u_1) + H(u_2) - H(u_1 | u_2). \tag{3.40}
\]

Using \( H(u) = H(u_2) + H(u_1 | u_2) \) and the definition of mutual information,

\[
I(u_1; u_2) \triangleq H(u_1) - H(u_1 | u_2), \tag{3.41}
\]

we get

\[
D(p_{u_1, u_2} \| p_{u_1} p_{u_2}) = I(u_1; u_2). \tag{3.42}
\]

For the blind separation problem, the random vector \( \mathbf{u} \) contains all \( M_s \) signals. Rather interestingly, the minimization of the Kullback-Leibler divergence of the joint output distribution and the product of the marginal output distributions, \( D(p_{u_1, u_2}, \ldots, p_{u_{M_s}}) \), leads to the same update equation, Eq. (3.16), but with a different nonlinearity. However, the nonlinearity is not as accessible as the one derived by the Infomax approach, as marginal entropies appear in the expression, which cannot be expressed in closed form, even if the output distributions were replaced by the source distributions. These marginal entropies can be expressed in truncated Gram-Charlier approximations of the marginal output distributions, which leads to a nonlinearity of the form

\[
g(u) = \frac{3}{4} u^1 + \frac{15}{4} u^9 - \frac{14}{3} u^7 - \frac{29}{4} u^5 + \frac{29}{4} u^3. \tag{3.43}
\]

3.6.4 The EASI algorithm

Cardoso and Laheld [26] have come up with an algorithm quite close to Eq. (3.17). Their equivariant adaptive separation via independence (EASI) algorithm has the form

\[
W_{t+1} = W_t + \mu \left( I - uu^T - g(u)u^T + ug(u)^T \right) W_t. \tag{3.44}
\]

A close look at Eq. (3.44) reveals that the EASI algorithm, also termed the symmetric form of ICA [24], will simultaneously decorrelate the signal, so that criteria can be packed into the nonlinearity that alone would not be sufficient. Cardoso [24] calls them constrained criteria. If the score functions are used for \( g(.) \), the properties of Eq. (3.44) are similar to that of Eq. (3.17). However, for model mismatch (the true pdf is unknown or different from the assumption), the stability region of the EASI algorithm can under certain circumstances be more tolerant than that of Eq. (3.17). As a disadvantage, there are more terms to evaluate in Eq. (3.44).

A hybrid form, which stands somewhere between Eq. (3.17) and Eq. (3.44), was given by Girolami and Fyfe [57]

\[
W_{t+1} = W_t + \mu \left( I - uu^T - g(u)u^T \right) W_t. \tag{3.45}
\]

where \( g_i(u_i) = \tanh(u_i) \) for super-Gaussian, and \( g_i(u_i) = -\tanh(u_i) \) for sub-Gaussian distributions, respectively, i.e., a simple sign change can account for a different class of distributions.

3.6.5 CMA and kurtosis

Another algorithm, the so-called multiuser constant-modulus algorithm (MU-CMA), which is derived from its blind deconvolution counterpart, was given by Papadias [109]. Similarly, multiuser kurtosis-based algorithms were developed.

In [5] Eq. (3.43) is claimed to be model independent, since the knowledge of the source distributions was not assumed deriving the nonlinearity. However, in Section 4.7 of this thesis it is shown that this nonlinearity cannot possibly separate all non-Gaussian distributions.
by different people, e.g., Cardoso [26]. Both criteria, the CMA and the kurtosis essentially need prewhitening or simultaneous cross-decorrelation; otherwise, several outputs might lock on to the same source.

Sufficient and necessary conditions (e.g., [120]) for BSS using the kurtosis as a criterion have been given (original or maximum kurtosis and decorrelation). Kung et al. [83] have introduced methods to recursively extract all source signals with nonzero kurtoses from a hybrid mixture of super- and sub-Gaussian sources.

### 3.6.6 Semiblind signal separation

In certain applications some of the source signals of a mixture are known, e.g., in systems with feedback. For example, in acoustic applications (e.g., teleconferencing) the output signal of a loudspeaker may leak into several microphones that are placed loosely in front of a group of speakers in the same room. Although the speakers’ signals have to be separated blindly, speaker separation can be accelerated if the known reference signal of the loudspeaker is incorporated into the blind algorithm.

Schobben and Sommen [118] showed a way of combining echo cancellation and blind separation, which increased both effectiveness in time-varying environments and computational efficiency. Three different ways of combining blind and nonblind signal separation methods are introduced by Joho et al. [73, 76]. The setup of these algorithms is depicted in Figs. 3.5–3.7 respectively.

![Figure 3.5: Combined echo canceler and blind signal separation.](image)

There are two update equations for the algorithm shown in Fig. 3.5. The nonblind part is given by

$$H_{t+1} = H_t + \mu e_b s^T_t,$$

(3.46)

whereas the blind update equation is either Eq. (3.17) or Eq. (3.18).

![Figure 3.6: Combined equalizer and blind signal separation.](image)

The update equation for the algorithm in Fig. 3.6 incorporates both blind and nonblind adaptation. The separation matrix is updated according to a variant of Eq. (3.18)

$$W_{t+1} = W_t + \mu (R_{ss} - R_{us}) (\hat{R}_{ss} - \mu (R_{ss} - R_{us}))^{-1} W_t,$$

(3.47)

Without loss of generality, the elements of the source signal vector $s$ might be arranged such that its first $M_0$ components are the unknown source signals and the remaining $M_r$ elements contain the source signals with known references, i.e., $s \triangleq [s_0 \ s_r]^T$. Likewise $u \triangleq [u_0 \ u_r]^T$. This simplifies the representation of the matrices and allows block notation. The matrix $\hat{R}_{ss}$ consists partly of an identity matrix and partly of an estimate of the cross-correlation matrix of the known source signals

$$\hat{R}_{ss} \triangleq \begin{bmatrix} I_{M_0} & 0 \\ 0 & \hat{R}_{ss,r} \end{bmatrix}.$$  

(3.48)

If the known source signals are mutually uncorrelated, $\hat{R}_{ss}$ is a diagonal matrix and simplifies the matrix inversion in Eq. (3.47) near convergence. The matrices $R_{ss}$ and $R_{us}$ contain running or instantaneous estimates of correlation terms. More precisely, they contain correlation information of the unknown
source signals and the reference sources as well as cross-correlation between the two. For $\sigma_{u_1}^2 = \ldots = \sigma_{u_{M_b}}^2 = 1$ they show the following block structure:

$$\mathbf{R}_{SS} \triangleq \begin{bmatrix} \mathbf{I}_{M_b} & \mathbf{0} \\ \mathbf{0} & \mathbf{s_r} \end{bmatrix}$$

(3.49)

and

$$\mathbf{R}_{US} \triangleq \begin{bmatrix} g(u_b) \cdot u_b^T & u_b \cdot \mathbf{s_r}^T \\ u_r \cdot u_b^T & u_r \cdot \mathbf{s_r}^T \end{bmatrix}.$$  

(3.50)

From Eq. (3.50) it can be seen that nonlinear functions are only needed for the signals where no reference is known. For the signals with a reference, correlations with the reference and correlations between blind and nonblind output signals are sufficient. In contrast to $\mathbf{R}_{SS}$ and $\mathbf{R}_{US}$, which basically contain blocks of outer products of two vectors for a given time instant $t$ and therefore are not of full rank, $\mathbf{R}_{SS}$ has to be nonsingular, i.e., its rank must be $M_b$, otherwise the inverse of $[\mathbf{R}_{SS} - \mu(\mathbf{R}_{SS} - \mathbf{R}_{US})]$ in Eq. (3.47) does not exist.

The third method introduces the notion of virtual sensors. We assume that the update equation is of the form

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \mu \Delta \mathbf{W}_t.$$  

(3.51)

Although $\mathbf{u}_r$ could be estimated blindly using Eq. (3.17), we already know that $\mathbf{u}_r = \mathbf{s}_r$, with $\mathbf{s}_r$ being the known subset of source signals. In the blind-only case, $M_s = M_b + M_r$ different mixtures of the source signals are required in order to completely separate all of them. But since $M_r$ source signals are known, we can set up $M_r$ virtual sensors, whose signals are the known source signals, see Fig. 3.7. The remaining $M_b$ real sensors contain different mixtures of all source signals (including the known ones). For the sensor signals we can therefore write

$$\mathbf{x} \triangleq \begin{bmatrix} \mathbf{x}_b \\ \mathbf{s}_r \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{s}_b \\ \mathbf{s}_r \end{bmatrix}.$$  

(3.52)

The structure of the mixing matrix $\mathbf{A}$ can now be represented in block form, revealing the dependence on the two subsets of signals. The matrices $\mathbf{A}_{11}$ and $\mathbf{A}_{12}$ describe the influence of the unknown and known source signals to the real sensor signals, respectively. The zero and the identity submatrix result from the introduction of the $M_r$ virtual sensors. The separation matrix $\mathbf{W}$ can now also be written as a simplified block matrix, hence

$$\mathbf{u} \triangleq \begin{bmatrix} \mathbf{u}_b \\ \mathbf{s}_r \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{s}_b \\ \mathbf{s}_r \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} + \mathbf{W}_{11} \mathbf{A}_{12} + \mathbf{W}_{11} \mathbf{A}_{12} \mathbf{W}_{12} + \mathbf{W}_{11} \mathbf{A}_{12} \mathbf{W}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{s}_b \\ \mathbf{s}_r \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \end{bmatrix} \mathbf{s}.$$  

(3.53)

The lower part of $\mathbf{W}$ is now directly obtained as $[\mathbf{0}, \mathbf{I}]$ without adaptation, making the calculation of $[\Delta \mathbf{W}_{11}, \Delta \mathbf{W}_{12}]$ in Eq. (3.51) redundant. Thus, only the upper subblocks $[\mathbf{W}_{11}, \mathbf{W}_{12}]$ of $\mathbf{W}$ need to be updated, i.e.,

$$[\Delta \mathbf{W}_{11}, \Delta \mathbf{W}_{12}] = \left( \mathbf{I} - \mathbf{g}(\mathbf{u}_b) \mathbf{u}_b^T \right) \mathbf{W}_{11} \left( \mathbf{I} - \mathbf{g}(\mathbf{u}_b) \mathbf{u}_b^T \right) \mathbf{W}_{12} - \mathbf{g}(\mathbf{u}_b) \mathbf{s}_r^T.$$

(3.54)

The first term in Eq. (3.54) has exactly the same form as in Eq. (3.17), where no source signals are known. The second term only influences the updating of $\mathbf{W}_{12}$, which describes the signal flow between the virtual sensors $\mathbf{x}_r = \mathbf{s}_r$ and the output $\mathbf{u}_b$. The term $\mathbf{g}(\mathbf{u}_b) \mathbf{s}_r^T$ is a measure of the dependence between the known source signals $\mathbf{s}_r$ and the blindly recovered signals $\mathbf{u}_b$. During adaptation, the expectation of this term is reduced causing $\mathbf{u}_b$ to become a linear mixture of only unknown source signals $\mathbf{s}_b$. The first term in Eq. (3.54) then performs the actual separation, such that the output signals in $\mathbf{u}_b$ become independent. Since the two terms in Eq. (3.54) have different tasks regarding the updating of the separation matrix $\mathbf{W} \triangleq [\mathbf{W}_{11}, \mathbf{W}_{12}]$, different step sizes can be assigned to them. The resulting update equation is

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \mu_1 \left( \mathbf{I} - \mathbf{g}(\mathbf{u}_b) \mathbf{u}_b^T \right) \mathbf{W}_t - \mu_2 \left[ \mathbf{0}, \mathbf{g}(\mathbf{u}_b) \mathbf{s}_r^T \right].$$  

(3.55)
Simulation results in [73] have shown that the performance of such semiblind approaches is very close to that of systems where only the unknown signals are involved.

### 3.7 Stability conditions

#### 3.7.1 Stability analysis for complex-valued signals

The update equation, Eq. (3.17), may be extended to account for complex signals by exchanging the transpose operator by the Hermitian, thus giving

\[ W_{t+1} = W_t + \mu \left( I - g(u)u^H \right) W_t \]  

(3.56)

Since most stability analyses in the literature, e.g., [3,25,26,131] only treat the real case, we extend this to the complex case in the following. We consider the global transfer function of a system

\[ P = WA. \]  

(3.57)

For the following derivation we will use two definitions:

\[ D_{ss^*} \triangleq \text{diag} \left( \left[ s_1 s_1^*, \ldots, s_M s_M^* \right] \right) \]  

(3.58)

and

\[ D_{g^*} \triangleq \text{diag} \left( \left[ g'(s_1), \ldots, g'(s_M) \right] \right), \]  

(3.59)

where

\[ g'(s_k) \triangleq \nabla_{s_k} g(s_k) = \frac{\partial g(s_k)}{\partial \text{Re } s_k} + j \frac{\partial g(s_k)}{\partial \text{Im } s_k} \]  

(3.60)

(see Appendix A.4). Neglecting any possible permutation of the separating solution, we can express the global transfer function close to an equilibrium by

\[ P = I + \Delta P. \]  

(3.61)

where \( \Delta P \) is a perturbation of the system. The estimated source signal vector is then

\[ u = s + \Delta P s. \]  

(3.62)

The update equation of the global transfer function can be obtained by post-multiplying the update equation according to Eq. (3.56) by the matrix \( A \)

\[ P_{t+1} = P_t + \mu (I - g(u)u^H) P_t. \]  

(3.63)

Using Eq. (3.62), Eq. (3.63) may be written as

\[ P_{t+1} = I + \Delta P_{t+1} \]

\[ I + \Delta P_{t+1} = I + \Delta P_{t} + \mu \left( I - g(s + \Delta P_{t} s)(s + \Delta P_{t} s)^H \right) (I + \Delta P_{t}). \]  

(3.64)

Using as a first-order approximation

\[ g(s + \Delta P) \approx g(s) + Dg \Delta P, \]  

(3.65)

we can write

\[ \Delta P_{t+1} = \Delta P_{t} + \mu \left( I - (g(s) + Dg \Delta P_{t} s)(s + \Delta P_{t} s)^H \right) (I + \Delta P_{t}). \]  

(3.66)

Applying the expectation operator to Eq. (3.66), using \( E \{ g(s) s^H \} = I \), and neglecting all but the first-order terms, we get

\[ E \{ \Delta P_{t+1} \} = E \{ \Delta P_{t} \} - \mu \left( E \{ Dg \Delta P_{t} D_{ss^*} \} + E \{ g(s) s^H \} E \left( \Delta P_{t}^H \right) \}. \]  

(3.67)

We now treat the diagonal and the off-diagonal terms separately. For the diagonal terms we get

\[ E \{ \Delta P_{t+1} \} = E \{ \Delta P_{t} \} - \mu \left( E \{ g'(s) s_i s_i^* \} E \{ \Delta P_{t \neq i} \} + E \{ g(s) s_i^* \} E \{ \Delta P_{t \neq i}^* \} \right). \]  

(3.68)

We do not care about the phase of the perturbation, but the magnitude must necessarily shrink over time. Computing the magnitude of Eq. (3.68) results to first order in

\[ (\text{Re} (E \{ \Delta P_{t+1} \}))^2 + (\text{Im} (E \{ \Delta P_{t+1} \}))^2 = (\text{Re} (E \{ \Delta P_{t} \}))^2 + (\text{Im} (E \{ \Delta P_{t} \}))^2 
- 2 \mu E \{ g'(s) s_i s_i^* \} (\text{Re} (E \{ \Delta P_{t \neq i} \}))^2 
- 2 \mu E \{ g(s) s_i^* \} (\text{Re} (E \{ \Delta P_{t \neq i} \}))^2 
- 2 \mu E \{ g'(s) s_i s_i^* \} (\text{Im} (E \{ \Delta P_{t \neq i} \}))^2 
+ 2 \mu E \{ g(s) s_i^* \} (\text{Im} (E \{ \Delta P_{t \neq i} \}))^2, \]  

(3.69)
from which we can derive the condition

\[ E \left[ g'(s_i)s_is_i^* \right] \left( \text{Re} \left( E \left[ \Delta p_{ji} \right] \right) \right)^2 \right. + E \left[ g(s_i)s_i^* \right] \left( \text{Re} \left( E \left[ \Delta p_{ji} \right] \right) \right)^2 \left. \right. + \left. \left. E \left[ g'(s_i)s_is_i^* \right] \left( \text{Im} \left( E \left[ \Delta p_{ji} \right] \right) \right)^2 \right. - E \left[ g(s_i)s_i^* \right] \left( \text{Im} \left( E \left[ \Delta p_{ji} \right] \right) \right)^2 > 0. \]  

(3.70)

Thus, sufficient conditions for the diagonal terms are

\[ E \left[ g'(s_i)s_is_i^* \right] > E \left[ g'(s_i) \right] E \left[ s_is_i^* \right], \]  

(3.71)

\[ E \left[ g'(s_i)s_is_i^* \right] - E \left[ g(s_i)s_i^* \right] > 0. \]  

(3.72)

Note that in the real-valued case, only Eq. (3.71) has to be satisfied. For monotonic odd nonlinearities, Eq. (3.71) is easily satisfied. Eq. (3.72), however, needs a closer look. For nonlinearities whose first derivative is convex, as is the case for most complex distributions related to digital communication constellations, we find that

\[ E \left[ g'(s_i)s_is_i^* \right] > E \left[ g'(s_i) \right] E \left[ s_is_i^* \right], \]  

(3.73)

a proof of which is provided in Appendix C.2, so that Eq. (3.72) will automatically be satisfied by the condition for the off-diagonal elements, Eq. (3.82), which follows further down. On the other hand, the off-diagonal term \( E \left[ \Delta p_{ji} \right] \) only depends on \( E \left[ \Delta p_{ji} \right] \) and \( E \left[ \Delta p_{ji} \right] \), and vice versa, so that we can formulate Eq. (3.67) as a 2 x 1 matrix equation

\[
\begin{bmatrix}
E \left[ \Delta p_{ji} \right] \\
E \left[ \Delta p_{ji} \right]
\end{bmatrix}
= 
\begin{bmatrix}
E \left[ \Delta p_{ji} \right] \\
E \left[ \Delta p_{ji} \right]
\end{bmatrix}
- \mu \begin{bmatrix}
E \left[ g'(s_i) \right] E \left[ s_j s_j^* \right] E \left[ \Delta p_{ji} \right] + E \left[ g(s_i) s_i^* \right] E \left[ \Delta p_{ji} \right] \\
E \left[ g(s_j) s_j^* \right] E \left[ \Delta p_{ji} \right] + E \left[ g'(s_j) \right] E \left[ s_j s_j^* \right] E \left[ \Delta p_{ji} \right]
\end{bmatrix}.
\]  

(3.74)

If we split the second vector on the RHS of Eq. (3.74) into real and imaginary parts of \( E \left[ \Delta p_{ji} \right] \) and \( E \left[ \Delta p_{ji} \right] \), we can write

\[
\begin{bmatrix}
E \left[ \Delta p_{ji} \right] \\
E \left[ \Delta p_{ji} \right]
\end{bmatrix}
= 
\begin{bmatrix}
E \left[ \Delta p_{ji} \right] \\
E \left[ \Delta p_{ji} \right]
\end{bmatrix}
- \mu \begin{bmatrix}
E \left[ g'(s_i) \right] E \left[ s_j s_j^* \right] E \left[ \Delta p_{ji} \right] \\
E \left[ g(s_j) s_j^* \right] E \left[ \Delta p_{ji} \right] + E \left[ g'(s_j) \right] E \left[ s_j s_j^* \right] E \left[ \Delta p_{ji} \right]
\end{bmatrix}.
\]  

(3.75)

Sufficient conditions for stability are obtained by treating the real and imaginary parts of the perturbation update individually. As a result, both matrices in Eq. (3.75) must have positive eigenvalues. We denote

\[
H = \begin{bmatrix}
a_i & b_i \\
b_j & a_j
\end{bmatrix}
\]  

and

\[
\tilde{H} = \begin{bmatrix}
a_i & -b_i \\
-b_j & a_j
\end{bmatrix}.
\]  

(3.77)

In fact, the two matrices \( H \) and \( \tilde{H} \) share the same eigenvalues \( \kappa_\pm \), which can be found by solving the characteristic equation of \( H \),

\[
\det (H - \kappa \mathbf{I}) = 0,
\]  

(3.78)

which leads to the solutions

\[
\kappa_\pm = \frac{a_i + a_j}{2} \pm \frac{a_i + a_j}{2} - b_i b_j - a_i a_j.
\]  

(3.79)

For both eigenvalues to be positive, we must have

\[
a_i a_j > b_i b_j.
\]  

(3.80)

Hence,

\[
E \left[ g'(s_i) \right] E \left[ s_i s_i^* \right] E \left[ g'(s_j) \right] E \left[ s_j s_j^* \right] E \left[ g(s_i) s_i^* \right] E \left[ g(s_j) s_j^* \right] E \left[ g(s_i) s_i^* \right] E \left[ g(s_j) s_j^* \right] E \left[ g(s_i) s_i^* \right] E \left[ g(s_j) s_j^* \right].
\]  

(3.81)

Thus, sufficient conditions for the individual nonlinearities are

\[
E \left[ g'(s_i) \right] E \left[ s_i s_i^* \right] > E \left[ g(s_i) s_i^* \right].
\]  

(3.82)

Eq. (3.82) also becomes apparent if considering all source distributions to be equal\(^6\). Thus, the eigenvalues become

\[
\kappa_+ = E \left[ g'(s) \right] E \left[ s s^* \right] + E \left[ g(s) s^* \right],
\]  

(3.83)

\[
\kappa_- = E \left[ g'(s) \right] E \left[ s s^* \right] - E \left[ g(s) s^* \right].
\]  

(3.84)

For both \( \kappa_+ \) and \( \kappa_- \) to be positive, again Eq. (3.82) has to be satisfied.

\(^6\)In such cases we allow ourselves to drop the index \( i \).
3.7.2 Stability regions of some nonlinearities

Although we have derived the stability conditions for complex signals, very often the conditions for real signals, which are a subset of the conditions for complex signals, are sufficient. In the following we list these conditions for some common nonlinearities. Note that this analysis concerns local stability, hence the statistics of \( s \) and \( u \) are interchangeable near convergence. The stability condition for real signals is easily derived from Eq. (3.82)

\[
E \{ g'(u) \} E \left[ u^2 \right] - E \{ g(u)u \} > 0. \tag{3.85}
\]

For monomial nonlinearities (power functions)

\[
g(u) = a |u|^{p-1}, \tag{3.86}
\]

the stability condition, Eq. (3.85), can be written in terms of the pdf \( p_\alpha \) of the corresponding normalized random variable \( \tilde{u} \), which is a scaled version of the original random variable \( u \). Thus, we have

\[
u = \sigma_u \tilde{u}. \tag{3.87}
\]

Using Eqs. (3.86) and (3.87) in Eq. (3.85) results in

\[
p \sigma_u^{p+1} a E \left[ \tilde{u}^{p-1} \right] - \sigma_u^{p+1} a E \left[ \tilde{u}^{p+1} \right] > 0, \tag{3.88}
\]

which is written in terms of the nonlinearities as

\[
E \{ g'(\tilde{u}) \} - E \{ g(\tilde{u})\tilde{u} \} > 0 \tag{3.89}
\]

or

\[
\frac{E \{ g'(\tilde{u}) \}}{E \{ g(\tilde{u})\tilde{u} \}} > 1. \tag{3.90}
\]

This basically means that the scaling of monomial nonlinearities does not affect the stability region, which is entirely defined by the exponent of the monomial and the normalized distribution. Note that such a conclusion is not generally true for polynomials. However, a similar simplification of the stability condition can be carried out for the sign function

\[
g(u) = a \text{ sign}(u). \tag{3.91}
\]

For continuous distributions we know that, under scalar multiplication of a unit-variance distribution, the mode of a pdf is inversely proportional to its standard deviation

\[
p_u(0) \sim \frac{1}{\sigma_u}, \tag{3.92}
\]

so the stability condition can be written as

\[
E \{ g'(u) \} \sigma_u^2 \frac{2 \sigma_u^2 p_u(0)}{E \{ g(\tilde{u})\tilde{u} \} E \{ \tilde{u}^2 \}} \frac{2 \sigma_u^2 \frac{1}{\alpha} p_\alpha(0)}{E \{ \tilde{u} \} E \{ \tilde{u} \} } > 1. \tag{3.93}
\]

On the other hand, for general nonlinear functions, if we scale the nonlinearity properly such that

\[
E \{ g(\tilde{u})\tilde{u} \} = 1, \tag{3.94}
\]

then the stability condition, Eq. (3.85), simplifies to

\[
E \{ g'(\tilde{u}) \} > 1. \tag{3.95}
\]

Note that Eq. (3.95) is conditioned on the scaling constraint, Eq. (3.94). However, it has to be pointed out that the scaling condition is not a necessary condition for stability. It merely ensures unit-variance output signals and simplifies the stability condition equation, albeit not necessarily its satisfaction.

The stability condition, Eq. (3.85), has been evaluated for frequently applied nonlinearities\(^7\) and the resulting stability regions are given in Table 3.3. For those nonlinear functions with two entries in the stability-condition column, the first one is an unconditional stability condition, whereas the second entry is conditioned on satisfying the scaling constraint.

As an example we have a closer look at the power function for generalized Gaussian distributions. By inserting the moments of the generalized Gaussian distributions in the stability equation of Table 3.3 we get the condition on \( p \) and \( \alpha \) as

\[
p \frac{\Gamma \left( \frac{p+1}{\alpha} \right) \beta^{p-1}}{\Gamma \left( \frac{p+2}{\alpha} \right) \beta^{p+1}} > 1 \tag{3.96}
\]

or

\[
p \frac{\Gamma \left( \frac{\frac{p}{\alpha}}{\alpha} \right) \Gamma \left( \frac{3}{\alpha} \right)}{\Gamma \left( \frac{p+2}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} \right)} > 1. \tag{3.97}
\]

\(^7\)Most of these nonlinearities are derived in Chapter 4 or in Chapter 5.
<table>
<thead>
<tr>
<th>Nonlinearity</th>
<th>Scaling condition</th>
<th>Stability condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$au^3$</td>
<td>$a = \frac{1}{\kappa_4 + 3}$</td>
<td>$\kappa_4 &lt; 0$</td>
</tr>
<tr>
<td>$au</td>
<td>u</td>
<td>^{p-1}$</td>
</tr>
<tr>
<td>$a \text{sign}(u)$</td>
<td>$a = \frac{1}{E[</td>
<td>u</td>
</tr>
<tr>
<td>$a \tanh(u)$</td>
<td>$a = \frac{1}{E[\tanh(</td>
<td>u</td>
</tr>
<tr>
<td>$a \text{sign}(u) \left(1 - e^{-\sqrt{2}</td>
<td>u</td>
<td>}\right)$</td>
</tr>
<tr>
<td>Threshold NL</td>
<td>$a = \frac{1}{2\int p_0(\theta) , d\theta}$</td>
<td>$\int p_0(\theta) , d\theta &gt; 1$</td>
</tr>
</tbody>
</table>

Table 3.3: Stability regions of some nonlinearities. $\hat{u}$ is a random variable with unit variance.

By rewriting Eq. (3.97) we realize that we must find combinations of $\alpha$ and $\rho$ such that

$$\frac{\Gamma\left(\frac{\alpha+2}{\alpha}\right)}{\rho \Gamma\left(\frac{\rho}{\alpha}\right)} < \Gamma\left(\frac{3}{\alpha}\right)$$

is satisfied. This is the case if $0 < \alpha < 2$ and $\rho < 1$ by Lemma 2.2 and for $\alpha > 2$ and $\rho > 1$ by Corollary 2.2, respectively. In other words, super-Gaussian signals can be separated by power functions using powers of less than one, while sub-Gaussian signals by power functions using powers greater than one.

### 3.8 Overdetermined signal separation

Up to this point we have considered only cases where the number of sensor points equals the number of sources. If fewer sensor signals are available, the problem gets harder and is termed underdetermined, as it is no longer possible to separate all source signals completely (partial signal separation) in a linear separation manner [41], unless some restriction is made on the transmitted alphabet.

On the other hand, if more sensors than sources are available, the noise suppression capability might be enhanced by using overdetermined separation techniques [16, 31, 74, 139]. This overdetermined situation is sometimes also referred to as the problem of undercomplete bases. It is overdetermined in the sense that if the source signals are of interest, more observations than necessary for the reconstruction of the original source signals are available. However, if referring to the linear algebra system of finding the separation matrix given the mixing matrix, the term underdetermined is used, since we have more unknowns than equations. In the communication literature, this situation is referred to as diversity reception. Naturally, the diversity gain is much higher when channel fading occurs. Nonetheless, an improvement is also possible in a static AWGN channel, particularly if some sensors exhibit low SNRs. We will now derive the optimum solution in the sense of the minimum mean square error between the original source signals and the separated signals. Let the separated signals be a linear combination of the mixed source signals plus white Gaussian noise as given in Eq. (3.2). In the following, the matrices $R_s$ and $R_n$ denote the correlation matrix of the source and the noise signals, respectively. Using the mean square error as the cost function, we can write down the cost to minimize as

$$J_{\text{MSE}} \triangleq E \left\{ (s - \mathbf{u})^H (s - \mathbf{u}) \right\}.$$  (3.99)

Using Eq. (3.2) and setting the cost derivative with respect to the separation matrix $\mathbf{W}$ to zero, we arrive after some simplifications at

$$\frac{\partial J_{\text{MSE}}}{\partial \mathbf{W}} = E \left\{ (-s s^H + W A s^H + W n n^H) \right\} = 0.$$  (3.100)

Solving Eq. (3.100) for the separation matrix $\mathbf{W}$ yields

$$\mathbf{W} = R_s A^H (A R_s A^H + R_n)^{-1}.$$  (3.101)

From Eq. (3.101) we can derive some special cases. Very often, the signal sources as well as the noise sources are i.i.d. In this case we can write $R_s = \sigma_s^2 \mathbf{I}$. Then, the separation matrix becomes

$$\mathbf{W} = \frac{1}{\sigma_s^2} A^H (A + R_n)^{-1}.$$
\( \sigma_x^2 I \) and \( R_x = \sigma_x^2 I \). Eq. (3.101) can then be written as

\[
W = A^H \left( AA^H + \frac{\sigma_x^2}{\sigma_e^2} I \right)^{-1}.
\]

Both Douglas [43] and Joho et al. [74] suggest a two-stage blind approach to solve the overdetermined signal separation problem. Fig. 3.8 illustrates the underlying model for this approach for the situation where we have two sources and five sensors. The matrix \( A \) is now no longer square but it transforms the original source signals into a higher number of mixtures. At this point, sensor noise—or measurement noise—is added. The signals then become the input to the algorithm. In a first stage—the preprocessing step—the original number of signals is retrieved by a principal component analysis (PCA). In principle, this might be any subspace decomposition technique, see for example [43], to extract a higher SNR mixture of the signals of interest. The resulting signals \( v \) are now treated as signals coming from virtual sensors, so that any ICA technique will separate the signals. Of course, noise is still present after the PCA step, albeit at lower levels, and needs to be addressed by the following stage. As a consequence of a nonsquare matrix \( A \), the matrix \( W_d \) will have the transposed dimension of \( A \). The ICA stage is represented by the square matrix \( W_s \) with the dimension of the original number of sources. Simulation results in [74] show that diversity gains close to the theoretical optimum of MMSE solutions are possible. Similar techniques for overdetermined blind signal separation of noisy mixtures using multistage approaches were introduced by Chevalier [30].

3.9. Other modifications

There exists a vast variety of other, related algorithms for the blind separation problem, of which we will list a small subset below:

- The neural network structures described above need not necessarily be feed-forward structures. Recurrent neural networks are possible and are described, for example, by Jutten and Hérault [77], and Amari and Cichocki [4].
- If the source signals get convolved at the same time as mixed, techniques generally called multichannel blind deconvolution both separate and deconvolve the signals [7, 75, 84].
- Other approaches to the blind separation problem apply iterative techniques, where the separation of the source signals happens one at a time. Such approaches are often called blind signal extraction [6, 35, 93].

3.10. Summary

Gradient-search algorithms help overcome the computational burden of a direct solution of an ML equation. These methods become particularly admissible if noise is neglected. Modified-gradient variations (natural gradient) considerably increase the convergence speed of such algorithms. Some criteria allow to incorporate knowledge on some of the source signals, leading to semiblind algorithms. A stability analysis for complex signals determines the viability of the presented algorithms for certain classes of distributions. The stability criterion offers a quantification of the amount of higher-order statistics implicitly produced by a nonlinearity. Overdetermined signal separation makes the optimum use of the signals available by combining the signals in order to achieve the highest diversity gain possible.
Chapter 4

Nonlinear functions

Nonlinear functions are a crucial part of many blind signal separation algorithms. Their role is defined by the objective or contrast function, which often is some kind of probabilistic or information-theoretical measure, such as likelihood, entropy, or mutual information, as was presented in the last chapter. In this chapter, in addition to employing a criterion and deriving the nonlinearity associated with it, we reverse this order and ask: “Given a certain nonlinear function, under what conditions is it a valid nonlinearity for the blind signal separation task?”

After some fundamental sections on Gaussian sources (Section 4.2), the scaling issues (Section 4.3), and the determination of convergence speed (Section 4.4), monomial functions, which are suitable for many real- and complex-valued signals, are introduced in Section 4.5.

The search for universally applicable nonlinearities in blind signal separation has produced nonlinearities that are optimal for a given distribution, as well as nonlinearities that are most robust against model mismatch. In Section 4.6 robustness is shown to be yet another justification for the score function. Section 4.7 provides a proof that a single, universal nonlinearity for the separation of all non-Gaussian signals cannot exist. It further introduces a class of difficult distributions, for which standard blind algorithms fail. Finally, Section 4.8 shows different methods of stabilizing BSS algorithms depending on the source distribution and the chosen nonlinearity, and compares their convergence performance.
4.1 Introduction

Throughout this thesis, if we speak of nonlinear functions, we will restrict ourselves to nonlinearities of the Bussgang type [14]. Bussgang nonlinearities are memoryless, i.e., their output is a function of the current input only. This leads to some statistical properties for the separated signal, in the sense that the expectation of the squared input to the Bussgang nonlinearity \( g() \) is equal to the expectation of the product of input and output of that Bussgang nonlinearity

\[
E[u^2] = E[g(u)u] \quad (4.1)
\]

or, for complex signals,

\[
E[u u^*] = E[g(u)u^*]. \quad (4.2)
\]

We will see in Section 4.3 how to scale a valid nonlinearity such that it becomes Bussgang.

In the past, very often successful computer simulations built the basis for claims that certain nonlinearities solve the BSS problem for certain classes of source distributions. Although the optimum nonlinearities (in terms of Fisher efficiency) for given source distributions were found by different researchers applying ML, Infomax, or other criteria (see last chapter), it was soon found that the exact curve of the nonlinearity did not matter too much [69]. This raises the question as to whether there are nonlinearities that can separate many different distributions, so-called universal nonlinearities. The search for a universal nonlinearity in the sense of a fixed nonlinearity for all non-Gaussian signals has been addressed by Amari [5].

4.2 Gaussian sources

The Gaussian distribution is known to be a difficult distribution for the blind separation problem. In fact, methods using higher-order statistics (explicitly or implicitly as the methods in this thesis do), fail completely when more than one Gaussian source is present. This is due to the equivalence of decorrelation and independence for Gaussian sources, which inhibits any true signal separation capability using HOS.

We now show that no nonlinearity can satisfy the stability condition, Eq. (3.82), for real-valued Gaussian distributions. This follows from Price’s theorem [110], which states that the covariance function for the input and the output of a nonlinearity for normal processes are related through

\[
R_{g(u)u}(\tau) = E \{g'(u)\} R_{uu}(\tau), \quad (4.3)
\]

where \( R_{g(u)u} \triangleq E \{g(u(t))u(t+\tau)\} \) and \( R_{uu} \triangleq E \{u(t)u(t+\tau)\} \) are cross- and autocorrelation functions, respectively. For \( \tau = 0 \), we get

\[
E[g(u)u] = E[g'(u)] E[u u], \quad (4.4)
\]

which is in clear violation to Eq. (3.82).

4.3 Scaling the nonlinearity

The score function is a Bussgang nonlinearity for the source pdf \( p_s() \) it was designed for, so that Eq. (4.1) or Eq. (4.2) applies. For scaled versions (different variance \( \sigma_v^2 \)), this may not be the case. However, the adaptive algorithm, Eq. (3.56), always makes

\[
E[g(u)u^*] = 1 \quad (4.5)
\]

by adapting \( \sigma_v^2 \) accordingly.

By the scaling ambiguity, which is an inherent property of blind signal separation (see Chapter 3), it is impossible to recover the original power of the source signals without further knowledge. When the original power of the source signals is unknown, it is reasonable to normalize the power after the separation matrix to

\[
E[uu^T] = I. \quad (4.6)
\]

This can be achieved by a separate automatic gain control stage (AGC) after the BSS algorithm, or by scaling the nonlinearity \( g() \) properly. We have seen in Chapter 3 that at an equilibrium point of Eq. (3.56) we have

\[
E[g(u)u^T] = I. \quad (4.7)
\]
Hence, for every component \( u = u_i \) of the vector \( u \), \( i = 1, \ldots, M \), we need to scale \( g(u) \) such that
\[
\int_{-\infty}^{\infty} p_s(u) g_i(u) u^i \, du = 1, \tag{4.8}
\]

where \( p_s(.) \) is a source distribution with \( \sigma_s^2 = 1 \). But because of Eq. (4.6), such a properly scaled nonlinearity is Bussgang. Note that for the score function, Eq. (3.15), of most common distributions, Eq. (4.8) is satisfied without further scaling. We can see this by inserting the score function, Eq. (3.15), of a real-valued random variable into Eq. (4.8) and obtaining (integration by parts)
\[
\int_{-\infty}^{\infty} -p_s^i(u) \, du = -p_s(u) u^i \bigg|_{-\infty}^{\infty} + \int_{-\infty}^{\infty} p_s(u) \, du = 1. \tag{4.9}
\]

An example shall illustrate the concept of properly scaling the nonlinearity if a function other than the score function is used. For sub-Gaussian signals, separation with the nonlinearity \( g(u) = a \cdot u^3 \) is common—although this nonlinearity is only the true score function for the generalized Gaussian distribution with the exponential parameter \( \alpha = 4 \)—so that the scaling condition equates to
\[
\int_{-\infty}^{\infty} p_s(u) a u^3 \, du = 1 \tag{4.10}
\]
or, solved for the gain factor,
\[
a = \frac{1}{\int_{-\infty}^{\infty} p_s(u) u^4 \, du}. \tag{4.11}
\]

With the definition of the kurtosis in mind (cf. Chapter 2), we see that
\[
a = \frac{1}{\kappa_4 + 3}. \tag{4.12}
\]

For the uniform distribution, a properly scaled nonlinearity is therefore
\[
g_{\text{uni}}(u) = \frac{5}{9} \cdot u^3. \tag{4.13}
\]

Likewise, we can write down the respective nonlinearities for \( M \)-PAM and \( M \)-QAM
\[
g_{\text{PAM}}(u) = \frac{5(M^2 - 1)}{9M^2 - 21} \cdot u^3, \tag{4.14}
g_{\text{QAM}}(u) = \frac{5(M - 1)}{7M - 13} \cdot u^3. \tag{4.15}
\]

In other words, the cubic nonlinearity can be utilized to separate uniformly distributed, PAM, or QAM signals, with the only difference of a scalar gain in order to produce unit-variance output signals.

### 4.4 Convergence speed of the BSS algorithm

Beside the satisfaction of the stability criterion, we would like to quantify the convergence behavior of Eq. (3.17) for a certain nonlinearity. Von Hoff et al. [131] have come up with a measure that estimates the steady-state performance, which is of course traded off against convergence speed via the step size \( \mu \) as with all adaptive algorithms [65]. Similarly to the definition of \( \kappa_+ \) and \( \kappa_- \), more nonlinear moments are defined according to
\[
\gamma_+ \triangleq E \left[ g^2(s) \right] E \left[ s^2 \right] + \left( E \left[ g(s) s \right] \right)^2, \tag{4.16}
\]
\[
\gamma_- \triangleq E \left[ g^2(s) \right] E \left[ s^2 \right] - \left( E \left[ g(s) s \right] \right)^2. \tag{4.17}
\]

The steady-state error can then be shown to be proportional to
\[
\xi \triangleq \frac{1}{2} \left( \frac{\gamma_+}{\kappa_+} + \frac{\gamma_-}{\kappa_-} \right). \tag{4.18}
\]

A possible goal is now the minimization of the misadjustment, which is proportional to \( \xi \), for stable nonlinearities, or even better the minimization of the ratio of misadjustment and smallest eigenvalue, which in turn expresses local convergence speed, hence \( \xi / \kappa_- \) as indicated by von Hoff [130]. This relates to techniques used to compare nonblind, LMS-like adaptive algorithms with different error functions [52, 104, 132].

### 4.5 Monomial functions

As mentioned earlier, the basic form of the nonlinearity depends on the distribution of the source signal. From Eq. (3.15) it can be seen, that super-Gaussian signals typically have sigmoid-like activation functions such as \( \text{sign}(\cdot) \) or \( \tanh(\cdot) \), while for sub-Gaussian distributed real signals, a nonlinearity of the form \( g(u) = a \cdot u^3 \) is often selected [7]. The value of \( a \) is determined...
by Eq. (4.8) (scaling constraint). Eq. (2.45) reveals immediately (by setting \( m = p + 1 \)) that the monomial function

\[
g(u) = a \cdot u^p, \quad p \text{ odd}, \quad p \geq 3
\]

(4.19)
is a stable nonlinearity for sub-Gaussian signals. An odd value of \( p \) ensures the validity of the sign after the nonlinearity. If Eq. (4.19) is rewritten as

\[
g(u) = a \cdot |u|^p, \quad p > 1,
\]

(4.20)
\( p \) is no longer restricted to odd integers, but can be any rational number greater than one. Eq. (4.20) also has the advantage that it is directly applicable to complex signals, as will be seen further down. If we take the generalized Gaussian distribution, we get the scaling factor \( a \) as follows

\[
a = \frac{1}{\int_{-\infty}^{\infty} s^p \, p_d(s) \, ds} = \frac{1}{E\{s^{p+1}\}} = \frac{\Gamma\left(\frac{1}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right)} \beta^{-(p+1)}. \tag{4.21}
\]

Furthermore, from the definitions given in Eqs. (3.83), (3.84), (4.16), and (4.17), we can compute

\[
\kappa_\pm = \frac{\Gamma\left(\frac{2}{a}\right) \Gamma\left(\frac{3}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right) \Gamma\left(\frac{1}{a}\right)} p \pm 1 \tag{4.22}
\]
and

\[
\nu_\pm = \frac{\Gamma\left(\frac{2p+3}{a}\right) \Gamma\left(\frac{3}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right) \Gamma\left(\frac{1}{a}\right)} 2 - p \pm 1. \tag{4.23}
\]

Using Eq. (4.18) we get

\[
\xi = \frac{\Gamma\left(\frac{1}{a}\right) \Gamma\left(\frac{2}{a}\right) \Gamma\left(\frac{3}{a}\right) \Gamma\left(\frac{p+1}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right) \Gamma\left(\frac{1}{a}\right) \Gamma\left(\frac{3}{a}\right) ^2} - \left(\Gamma\left(\frac{p+2}{a}\right) \Gamma\left(\frac{1}{a}\right) \right)^2
\]

(4.24)
which the steady-state error of BSS algorithms using the monomial nonlinearity is proportional to for generalized Gaussian source distributions.

The best-known case of a sub-Gaussian distribution is the uniform. It can be regarded as a special case of the generalized Gaussian distribution with \( \alpha \to \infty \). Hence, the scaling can be deduced from Eq. (4.21). Other sub-Gaussian distributions include discrete distributions often used in communications, which can be modeled in a first approximation by uniform distributions, particularly when the number of discrete levels is large. One such distribution is \( M \)-ary pulse amplitude modulation (PAM) (also called \( M \)-PAM or \( M \)-ASK). The correct scaling for all these distributions is summarized in Table 4.1.

<table>
<thead>
<tr>
<th>distribution</th>
<th>scaling factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>gen. Gaussian (sub-)</td>
<td>( a = \frac{\Gamma\left(\frac{1}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right)} \left(\frac{\Gamma\left(\frac{1}{a}\right)}{\Gamma\left(\frac{p+2}{a}\right)}\right)^{\frac{p+1}{2}} )</td>
</tr>
<tr>
<td>uniform</td>
<td>( a = \frac{p+2}{3^{\frac{p+1}{2}}} )</td>
</tr>
<tr>
<td>( M )-PAM</td>
<td>( a = \frac{M}{2} \left(\frac{M^2-1}{3}\right)^{\frac{p+1}{2}} \sum_{m=1}^{M/2} (2m-1)^{p+1} )</td>
</tr>
<tr>
<td>general</td>
<td>( a = \frac{1}{E{u^{p+1}}} )</td>
</tr>
</tbody>
</table>

**Table 4.1:** Scaling factors for monomial nonlinearities according to Eq. (4.20). Distributions are assumed real and to have unit variance.

### 4.5.1 QAM signals

\( M \)-ary quadrature amplitude modulation is an important digital modulation format, which can be regarded as an extension of pulse amplitude modulation to the complex baseband representation. For the common case that \( M \) is an even power of two, e.g., \( M = 16 \), the I and Q parts are independent of each other. The distribution is sub-Gaussian.
4.5.2 Monomials for complex baseband signals

If the adaptive blind algorithm according to Eq. (3.56) is applied to complex-valued distributions, there are two distinct groups of nonlinearities to choose from:

- phase-preserving NLs,
- I/Q-independent NLs.

The question which group should be applied not only depends on the achievable performance but also on the requirements to other synchronization tasks. If carrier recovery is relegated to a later stage in the receiver chain, phase-preserving nonlinearities (e.g., CMA) have to be employed.

Phase-preserving nonlinearities, as their name suggests, leave the phase of the input signal unchanged. Such nonlinearities have the form \( f(|u|) \cdot u \). Any rotation left in the matrix \( W \) at the equilibrium will be conserved, since

\[
g(u)u^* = f(|u|)u^* = f(|u|)|u|^2
\]

is always real, so that \( E\{1 - g(u)u^*\} = 0 \) is independent of the phase of signal \( u \). Eq. (4.20) can thus be used for all complex-valued sub-Gaussian signals. The respective scaling factors can be obtained from Table 4.2.

<table>
<thead>
<tr>
<th>distribution</th>
<th>scaling factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>M-PSK</td>
<td>( a = 1 )</td>
</tr>
<tr>
<td>M-QAM</td>
<td>( a = \frac{M}{4} \left( \frac{2(M-1)}{3} \right)^{2^{M-1}} \sum_{m=0}^{M/2-1} \sum_{n=0}^{M/2-1} \left( (2m-1)^2 + (2n-1)^2 \right)^{2^{M-1}} )</td>
</tr>
<tr>
<td>general</td>
<td>( a = \frac{1}{E{</td>
</tr>
</tbody>
</table>

Table 4.2: Scaling factors for monomial nonlinearities according to Eq. (4.20) (phase preserving) for complex-valued, unit-variance distributions.

I/Q-independent nonlinearities, on the other hand, apply the same nonlinearity independently to the real and the imaginary parts of the signal \( u = u_R + j u_I \). Such a nonlinearity is strongly motivated by the interpretation of the score function to a distribution whose real and imaginary parts are independent. By writing the derivative as a complex gradient (see Appendix A.4) we have

\[
g(u) = \frac{\nabla_u p_u(u)}{p_u(u)} = \frac{\frac{\partial p_u(u)}{\partial u_R}}{p_u(u)} + j \frac{\frac{\partial p_u(u)}{\partial u_I}}{p_u(u)} \tag{4.26}
\]

For independent in-phase and quadrature components, we have

\[
p_u(u) = p_{u_R}(u_R) \cdot p_{u_I}(u_I). \tag{4.27}
\]

Hence, the nonlinearity becomes

\[
g(u) = -\frac{\frac{\partial p_{u_R}(u_R)}{\partial u_R} p_{u_I}(u_I) + j \frac{\partial p_{u_I}(u_R)}{\partial u_I} p_{u_R}(u_R)}{p_{u_R}(u_R) p_{u_I}(u_I)}
\]

Nonlinearities of the form given by Eq. (4.29) rotate the constellation of a complex signal such that the real and imaginary parts become independent of each other. Equilibrium now means that

\[
E\{1 - g(u)u^*\} = E\{1 - (g(u_R) + j g(u_I))(u_R - j u_I)\}
\]

Two equations can be built from this, namely

\[
E\{g(u_R)u_R - g(u_I)u_I + j (g(u_R)u_I - g(u_I)u_R)\} = 0 \tag{4.30}
\]

and

\[
E\{g(u_R)u_I + g(u_I)u_R\} = 1 \tag{4.31}
\]

Eq. (4.31) makes sure that the signals are separated by using both real and imaginary parts. Eq. (4.32) enforces a further criterion on the real and imaginary parts.
parts, making the constellation rotate. Furthermore, all the cross-correlation terms between different source signals are still forced to zero by

\[ E \left\{ g(u_{ik})u_{i,k} + g(u_{il})u_{l,k} \right\} = 0, \quad i \neq k, \]  

(4.33)

\[ E \left\{ g(u_{ik})u_{i,k} - g(u_{il})u_{l,k} \right\} = 0, \quad i \neq k. \]  

(4.34)

As the nonlinearities are individual for real and imaginary parts, so are the scaling factors. Thus, M-QAM signals are in fact scaled like \( \sqrt{M}\)-PAM signals with an additional factor of \( 1/2 \) due to Eq. (4.31), and a slightly different numerator in the expression for \( a \) due to the normalization of a complex constellation. Some results are collected in Table 4.3.

<table>
<thead>
<tr>
<th>distribution</th>
<th>scaling factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>M-QAM</td>
<td>( a = \frac{\sqrt{M}}{\sqrt{M/2} \sum_{m=1}^{M} (2m-1)^{p+1}} )</td>
</tr>
<tr>
<td>general</td>
<td>( a = \frac{1}{2E{u_{k}^{p+1}}} = \frac{1}{2E{u_{k}^{p+1}}} )</td>
</tr>
</tbody>
</table>

Table 4.3: Scaling factors for monomial nonlinearities according to Eq. (4.29) \((I/Q \text{ independent})\). Distributions are assumed to have unit variance.

### 4.6 Optimal nonlinearities

#### 4.6.1 The form of the nonlinearity

If the separation of signals of a certain class of distributions is the goal, the literature suggests to apply nonlinearities of the form \( g(u) = au^{3} \) for sub-Gaussian signals and \( g(u) = a \tanh(bu) \) for super-Gaussian signals, where \( a \) is a scalar used to adjust the output power. These nonlinearity choices can be refined according to the stability conditions given earlier, as summarized in Table 3.3.

An intuitive explanation of the appropriate form of the nonlinearity can be given as follows. If the nonlinearity is properly scaled, i.e., \( E\{g(\hat{u})\hat{u}\} = 1 \), the stability condition \( E\{g'(\hat{u})\hat{u}\} > 1 \) determines if the separating points of the nonlinearity are locally stable. To ensure stability we aim at making \( E\{g'(\hat{u})\hat{u}\} \) as large as possible. For peaky distributions (super-Gaussian) where a large portion of the pdf lies around zero, the derivative of \( g(\cdot) \) should be large around this value, whereas with a flatter distribution, the contrary is the case. This means that super-Gaussian distributions need sigmoid-looking nonlinearities for their separation, which are concave functions for their arguments greater than zero, while sub-Gaussian distributions need nonlinearities of the form \( g(u) = uu^{p-1} \) with \( p > 1 \), showing a convex shape for \( u > 0 \).

Because sub-(super-) Gaussian signals have a negative (positive) kurtosis \( \kappa_4 \), these expressions are often used interchangeably, although the inverse direction of reasoning is not strictly applicable. Since the nonlinearities for super-Gaussian signals, e.g., \( \text{sign}() \), \( a \tanh() \), do not exhibit stability for the entire positive kurtosis plane, distributions might be constructed, for which both nonlinearities \( g(u) = au^{3} \) and \( g(u) = a \tanh(bu) \) fail [44].

#### 4.6.2 Optimization of the nonlinearity

The fact that the stability of blind separation algorithms depends on a nonlinear moment being greater than one implies that robustness of the algorithms can be obtained by making this nonlinear moment as large as possible [98]. Thus, we wish to maximize the left-hand side of the stability condition for a scaled nonlinearity according to Eq. (3.95). The scaling constraint

\[ \int_{-\infty}^{\infty} g(u) p_{i}(u) du = 1 \]  

(4.35)

alone is not sufficient. Any even part of \( g(\cdot) \) would show up neither in the constraint nor in the integral to maximize. But clearly, due to symmetry we wish to restrict \( g(\cdot) \) to odd functions. The optimization problem can be formulated as follows:

\[ \text{maximize} \quad \int_{-\infty}^{\infty} g'(u) p_{i}(u) du \]  

(4.36)

subject to \( \int_{-\infty}^{\infty} g^{2}(u) p_{i}(u) du = c, \)  

(4.37)
where $c$ is a constant. Now an even part of $g(\cdot)$ would increase the constraint unnecessarily without contributing to the integral to maximize. We are attempting to find the optimal nonlinearity by calculus of variations. To this end, we define

$$f \triangleq g'(u)p_s(u) + \lambda \left(g^2(u)p_s(u)\right), \quad (4.38)$$

where $\lambda$ is a Lagrange multiplier. To find the optimal $g(u)$, we have to solve the Euler-Lagrange equation [21]

$$\frac{\partial f}{\partial g} - \frac{d}{du} \frac{\partial f}{\partial g'} = 0,$$  

$$\frac{\partial}{\partial u} \frac{\partial f}{\partial g'} - \frac{\partial}{\partial g} \frac{\partial f}{\partial g'} - \frac{\partial}{\partial g'} \frac{\partial f}{\partial g''} = 0, \quad (4.39)$$

where we abridged $p \triangleq p_s(u)$, $p' \triangleq p'_s(u) = \frac{\partial}{\partial u} p_s(u)$, $g \triangleq g(u)$, and $g' \triangleq g'(u) = \frac{\partial}{\partial u} g(u)$. Working out the different terms of Eq. (4.39) for Eq. (4.38) yields

$$\frac{\partial f}{\partial g} = 2\lambda gp,$$  

$$\frac{\partial}{\partial u} \frac{\partial f}{\partial g'} = p',$$  

$$\frac{\partial}{\partial g} \frac{\partial f}{\partial g'} g' = 0,$$  

$$\frac{\partial}{\partial g'} \frac{\partial f}{\partial g''} g'' = 0.$$  

Using Eqs. (4.40)–(4.43) in Eq. (4.39) results in

$$g = \frac{1}{2\lambda} \frac{p'}{p}.$$  

The Lagrange multiplier $\lambda$ can now be found by the constraint on the output power of the nonlinearity. For that we would have to determine the constant $c$. Alternatively, we know that a further constraint is the one given originally. Inserting the solution Eq. (4.44) into Eq. (4.35) gives us

$$\int_{-\infty}^{\infty} \frac{1}{2\lambda} \frac{p'_s(u)}{p_s(u)} u p_s(u) du = \frac{1}{2\lambda} \int_{-\infty}^{\infty} u p'_s(u) du = 1.$$  

Integrating by parts yields for the integral in Eq. (4.45)

$$\int_{-\infty}^{\infty} u p'_s(u) du = p_s(u)u \bigg|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} p'_s(u) du = -1.$$  

The desired solution is thus $\lambda = -\frac{1}{2}$, leading to

$$g(u) = -\frac{p'_s(u)}{p_s(u)}, \quad (4.47)$$

which is exactly the score function. This is a further justification for the score function in addition to the ones already known, such as ML and Infomax.

### 4.7 A proof of the nonexistence of a universal nonlinearity

#### 4.7.1 The polynomial nonlinearity

Model-independent nonlinearities such as the one given in Eq. (3.43) have been claimed to separate both super- and sub-Gaussian distributions [5]. An intuitive explanation of why such approaches fail has been offered by Bell [10], by noticing that the corresponding function to which the nonlinearity is the score function is of the wrong type. Indeed, the evaluation of Eq. (3.43) by Eq. (3.85) reveals its failure. In the following a proof is given that such polynomial nonlinearities are bound to fail [97].

#### 4.7.2 Problem statement

We wish to prove that there is no general nonlinear function $g(\cdot)$ that is stable in the sense that

$$\sigma^2 \mathbb{E}\left[ g'(x) \right] > 1,$$  

for both a sub- and super-Gaussian distributed random variable. In the following, we restrict ourselves to the family of generalized Gaussian signals. This allows us to use some properties of higher-order moments as given in Chapter 2. The restriction does not result in a loss of generality, since if we can prove that no single nonlinearity can separate all sub- and super-Gaussian signals, then this result also means that no single nonlinearity can separate any non-Gaussian signals.
4.7.3 Stability analysis

Assume that the nonlinearity \( g(\cdot) \) is either an odd polynomial or that we can replace it by its Taylor series (with odd powers only)

\[
g(x) = \sum_{k \text{ odd}} t_k x^k. \tag{4.49}
\]

The Taylor series used in Eq. (4.49) assumes the existence of all derivatives of \( g(\cdot) \). If the nonlinearity is not smooth, however, the use of a polynomial can still make sense, since the expectation operator allows discontinuities in the nonlinearity, since its evaluation involves an integration over such singularities. Clearly, differentiation of \( g(x) \) w.r.t. \( x \) yields

\[
g'(x) = \sum_{k \text{ odd}} t_k k x^{k-1}. \tag{4.50}
\]

If Eqs. (4.49) and (4.50) are inserted into Eq. (4.48), we get the stability condition expressed as a function of the Taylor series coefficients

\[
\sigma_x^2 \sum_{k \text{ odd}} t_k E \{x^{k-1}\} \cdot \frac{\sum_{k \text{ odd}} t_k E \{x^{k+1}\}}{\sum_{k \text{ odd}} t_k E \{x^{k+1}\}} > 1. \tag{4.51}
\]

Because of Eq. (2.47) we now see that for Gaussian variables any nonlinearity will lead to

\[
\frac{\sigma_x^2 E \{g'(n)\}}{E \{ng(n)\}} = 1 \neq 1, \tag{4.52}
\]

which was of course to be expected. However, for strictly nonnegative values of \( t_k \) and \( x \) a sub-Gaussian variable, stability is guaranteed due to Eq. (2.51) and hence

\[
\sum_{k \text{ odd}} t_k E \{x^{k+1}\} < \sum_{k \text{ odd}} t_k k E \{x^{k-1}\} \sigma_x^2. \tag{4.53}
\]

In the following we will show that no polynomial nonlinearity can be found to separate both sub- and super-Gaussian signals. We assume that the signals have zero mean and symmetric distributions, leading to anti-symmetric nonlinearities.

**Theorem 4.1**

For the natural-gradient ICA update equation, Eq. (3.17), there does not exist a single fixed nonlinearity \( g(\cdot) \) that separates arbitrary mixtures of sub- and super-Gaussian signals.

**Proof:** We carry out this proof by induction. First we note that with only \( t_1 \neq 0 \) we have a linear function for \( g(\cdot) \), which is of course unable to separate any distribution. We will therefore have to add at least one further coefficient \( t_k \) unequal to zero. We then show that the choice of this coefficient is contradictory and hence cannot lead to stability. By induction, we can add as many coefficients as we like, but we will never reach stability for all distributions. The basis of the induction is to show that

\[
g(x) = t_1 x + t_3 x^3 \tag{4.54}
\]

cannot separate both sub- and super-Gaussian signals, because

\[
G(x) \triangleq \frac{\sigma_x^2 (t_1 + 3t_3 \sigma_x^2)}{\sigma_x^2 t_1 + t_3 E \{x^4\}} \tag{4.55}
\]

is always smaller than one either for sub-Gaussian or for super-Gaussian signals. We first note that \( t_1 \) and \( t_3 \) need to have different signs, otherwise \( G(x) \) is smaller than one for super-Gaussian signals, as can be easily verified using Eq. (2.50). Furthermore, we may restrict ourselves to positive values of \( t_1 \), implicating negative values of \( t_3 \), since the reverse case leads to identical values of \( G(x) \), as both numerator and denominator are linear expressions in both \( t_1 \) and \( t_3 \). We now distinguish between two cases.

Case 1: \( t_1 < -3t_3 \sigma_x^2 \). This makes the numerator of \( G(x) \) negative. For super-Gaussian signals the denominator is negative, too, but smaller than the numerator due to Eq. (2.50) and negative \( t_3 \). Thus, \( G(x) < 1 \) for super-Gaussian signals.

Case 2: \( t_1 > -3t_3 \sigma_x^2 \). Here we have a positive numerator. The denominator is greater than the numerator for sub-Gaussian signals due to Eq. (2.51) and negative \( t_3 \). Hence, \( G(x) < 1 \) for sub-Gaussian signals.

Suppose that a nonlinearity with only terms up to order \( K \) does not yield a stable solution. In the following we assume that a solution with

\[
g(x) = \sum_{k=1}^{K} t_k x^k \tag{4.56}
\]
will be stable for either sub- or super-Gaussian signals but not both. We now show that a solution that is unstable for either signal distribution class cannot in one step be extended to an everywhere stable solution. To stabilize the nonlinearity for all distributions, we add a further term in the nonlinearity: $t_{K+2} x^{K+2}$. This results in the extended nonlinearity

$$g(x) = \sum_{k=1}^{K+2} t_k x^k.$$

The stability criterion can now be written as

$$\sigma_x^2 \left( \sum_{k=1}^{K+2} t_k E \left[ x^{k-1} \right] + t_{K+2} (K+2) E \left[ x^{K+1} \right] \right) \geq 1.$$  \hspace{2cm} (4.57)

If we want to multiply both sides of Eq. (4.58) by the denominator, we have to distinguish two cases depending on the sign of the denominator of Eq. (4.58). Case 1 (positive denominator of Eq. (4.58)): From Eq. (4.58) it follows

$$t_{K+2} (\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]) > - \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right).$$

(4.59)

But due to Eq. (2.51),

$$\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right] > 0 \hspace{2cm} (4.60)$$

for sub-Gaussian signals, hence we have

$$t_{K+2} > - \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right) \frac{\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]}{\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]}.$$  \hspace{2cm} (4.61)

Along the same line of argument as in Case 1, we notice again the contradiction for the choice of $t_{K+2}$. We have now shown that if a nonlinearity

$$g(x) = \sum_{k=1}^{K} t_k x^k$$

so

$$t_{K+2} < \frac{- \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right)}{\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]},$$

(4.63)

In the limit for $\alpha \to 2$ for generalized Gaussian distributions, the right-hand sides of Eqs. (4.61) and (4.63) will become equal, making it impossible for $t_{K+2}$ to satisfy both conditions. The choice of $t_{K+2}$ restricts the stable range of the distribution parameter $\alpha$.

Case 2 (negative denominator of Eq. (4.58)): According to Eq. (4.58)

$$t_{K+2} (\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]) < - \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right).$$

(4.64)

Since

$$\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right] > 0$$

(4.65)

for sub-Gaussian signals, we have

$$t_{K+2} < \frac{- \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right)}{\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]}.$$  \hspace{2cm} (4.66)

But for super-Gaussian signals

$$\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right] < 0, \hspace{2cm} (4.67)$$

so

$$t_{K+2} > \frac{- \sum_{k=1, \text{odd}}^{K} t_k \left( \sigma_x^2 k E \left[ x^{k-1} \right] - E \left[ x^{k+1} \right] \right)}{\sigma_x^2 (K+2) E \left[ x^{K+1} \right] - E \left[ x^{K+3} \right]}.$$  \hspace{2cm} (4.68)
is capable of only separating either sub- or super-Gaussian signals (but not both), then its extended version, having an additional \((K+2)\)th term

\[
g(x) = \sum_{k \leq K+2} \beta_k x^k.
\]  

(4.70)

will also only separate either sub- or super-Gaussian signals (but not necessarily the same distribution class as the original nonlinearity), because \(\beta_{K+2}\) would have to be chosen greater and smaller than a certain value at the same time. By induction, this applies to any arbitrarily long odd power series.

\[\square\]

### 4.7.4 Difficult distributions

In the previous section we have seen that no fixed nonlinearity can separate all non-Gaussian distributions. In this section we show that the pair of the two most prominent nonlinearities for blind signal separation, namely the hyperbolic tangent and the cubic nonlinearity are not always able to separate a distribution either. In other words, there are distributions for which neither the hyperbolic tangent nor the cubic nonlinearity succeeds in separating. From Table 3.3 it becomes clear, that if a non-Gaussian distribution exists, that is neither separable by \(g(u) = u^3\) nor by \(g(u) = a \tanh(u)\), it has to show a positive kurtosis, since \(g(u) = u^3\) covers all negative-kurtosis distributions, but the stability region of \(g(u) = a \tanh(u)\) does not include all positive-kurtosis distributions.

One such peculiar distribution was given by Douglas [44]. It is a symmetric, discrete, quaternary signaling scheme with symbols \(\{\pm A_1, \pm A_2\}\), where \(A_2 = 3.8 A_1\) and \(P(x = A_2) = 0.035\). \(A_1\) is adjusted for unit variance resulting in \(A_1 = 0.718\). The kurtosis of this distribution is \(\kappa_4 = 1.12\). As can be checked, this distribution does not satisfy the stability condition for any of the two nonlinearities. In fact, we can create more of those “challenging” distributions (see Fig. 4.1) by making the following considerations. We only consider quaternary symmetric signals. With more levels of signaling, it is of course possible to create such critical distributions, too. However, using four symmetric levels provides enough degree of freedom to construct the desired distributions. We are looking at combinations of \(A_1, A_2, p_1 = P(x = A_1)\), and \(p_2 = P(x = A_2)\) that result in instability for both \(g(u) = u^3\) and \(g(u) = a \tanh(u)\). We assume that \(a\) is adjusted for unit-variance outputs. We have the following constraints:

\[\begin{align*}
p_1 + p_2 &= \frac{1}{2}, \\
p_1 A_1^2 + p_2 A_2^2 &= \frac{1}{2}, \\
p_1 A_1^4 + p_2 A_2^4 &\geq \frac{3}{2},
\end{align*}\]

(4.71)

(4.72)

(4.73)

From Eq. (4.71) and Eq. (4.72) we can express \(p_1\) and \(A_1\) as a function of \(p_2\) and \(A_2\)

\[\begin{align*}
p_1 &= \frac{1}{2} - p_2, \\
A_1^2 &= \frac{1 - 2p_2 A_2^2}{1 - 2p_2}.
\end{align*}\]

(4.74)

(4.75)

(4.76)
with the additional constraints

\[ p_2 \leq \frac{1}{2}, \quad A_2 \geq 1. \]  \hspace{2cm} (4.77)

From Eq. (4.73) and Eq. (4.77) we get a lower and an upper bound for \( p_2 \)

\[ p_2 \geq \frac{1}{A_2^2 - 2A_2^2 + 3}, \]  \hspace{2cm} (4.78)

\[ p_2 \leq \frac{1}{2A_2^2}. \]  \hspace{2cm} (4.79)

Invoking Eq. (4.74) we get an additional inequality for \( p_2 \) and \( A_2 \)

\[ 1 - (1 - 2p_2) \tanh^2 \left( \sqrt{\frac{1 - 2p_2A_2^2}{1 - 2p_2}} \right) - 2p_2 \tanh^2(A_2) \]

\[ > (1 - 2p_2) \sqrt{\frac{1 - 2p_2A_2^2}{1 - 2p_2}} \tanh \left( \sqrt{\frac{1 - 2p_2A_2^2}{1 - 2p_2}} \right) + 2p_2A_2 \tanh(A_2). \]  \hspace{2cm} (4.80)

The possible range of \( A_2 \) and \( p_2 \) is depicted in Fig. 4.2. One example of a “challenging” distribution can be extracted from Fig. 4.2 as \( A_2 = 5 \), \( p_2 = 0.005 \) and therefore \( p_1 = 0.495 \) and \( A_1 = 0.87 \).

In the next chapter we show that the threshold nonlinearity is able to separate such “challenging” distributions.

### 4.8 Stabilizing blind separation algorithms

We conclude this chapter by listing some possible solutions to the situation when the stability analysis of Eq. (3.17) reveals instability of the algorithm for a given distribution. If we assume all source signals to be of the same distribution, there is a selection of possibilities to stabilize the algorithm:

- change the nonlinearity,
- utilize the transpose property,
- change the sign in the extended Infomax or EASI.

The first option is a simple replacement of the nonlinearity by one whose stable region contains the distribution under consideration. Most often \( \tanh(.) \) is exchanged against \( u^3 \) and vice versa. The transpose property [131] as a second method of stabilization is a direct consequence of the stability analysis. If a term \( g(u)u^T \) previously resulted in instability, it can be stabilized by taking its transpose, \( ug(u)^T \). This can be applied to the ICA, the extended Infomax, or the EASI. A third possibility of stabilizing an update equation is by the change of a sign (or two in the case of the EASI) in the extended Infomax or EASI update equation.

The options to stabilize an algorithm can be combined, whereby an even number of options cancels. Combinations of modifications yielding stable update equations are listed in Table 4.4. The convergence properties for some of the unreferenced update terms in this table have not been proven. Their stable behavior for the example given is only indicated by the simulation. All simulations were carried out using ten sources and the step sizes adjusted such as to reach \( J_{\infty}(P) = -20 \text{dB} \). The results for super- and sub-Gaussian signals can be viewed in Figs. 4.3 and 4.4, respectively. The fastest algorithms are the EASI and the asymmetric ICA.
<table>
<thead>
<tr>
<th>distr.</th>
<th>nonlinearity</th>
<th>update term</th>
<th>ref.</th>
<th>curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>super-Gauss.</td>
<td>( g(u) = \tanh(u) )</td>
<td>( \Delta W_{t+1} = (I - g(u)u^T)W_t )</td>
<td>[3]</td>
<td>a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - g(u)u^T - uu^T)W_t )</td>
<td>[87]</td>
<td>b</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - g(u)u^T + ug(u)^T - uu^T)W_t )</td>
<td>[26]</td>
<td>c</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I + ug(u)^T - uu^T)W_t )</td>
<td></td>
<td>d</td>
</tr>
<tr>
<td>sub-Gauss.</td>
<td>( g(u) = u^3 )</td>
<td>( \Delta W_{t+1} = (I - u g(u)^T)W_t )</td>
<td>[40]</td>
<td>e</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - u g(u)^T - uu^T)W_t )</td>
<td></td>
<td>f</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I + g(u)u^T - uu^T)W_t )</td>
<td></td>
<td>g</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - u g(u)^T)W_t )</td>
<td>[7]</td>
<td>h</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - u g(u)^T - uu^T)W_t )</td>
<td></td>
<td>i</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - g(u)u^T + ug(u)^T - uu^T)W_t )</td>
<td>[26]</td>
<td>k</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I + g(u)u^T - uu^T)W_t )</td>
<td>[131]</td>
<td>l</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I - u g(u)^T - uu^T)W_t )</td>
<td></td>
<td>m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I + g(u)u^T - uu^T)W_t )</td>
<td>[87]</td>
<td>n</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \Delta W_{t+1} = (I + g(u)u^T - uu^T)W_t )</td>
<td></td>
<td>o</td>
</tr>
</tbody>
</table>

*Table 4.4: Stable choices of nonlinearities and update equations for adaptive blind signal separation.*

---

3 Strictly speaking, they applied the signum function as their nonlinearity, which is close in shape and stability properties to the hyperbolic tangent.

---

**Figure 4.3:** Convergence of the stable algorithms for BSS according to Table 4.4 for super-Gaussian (here Laplacian) signals.

**Figure 4.4:** Convergence of the stable algorithms for BSS according to Table 4.4 for sub-Gaussian (here uniform) signals.
4.9 Summary

The score function is a robust choice for model mismatch as long as the kurtosis sign does not change. The single, universal nonlinearity, however, which separates mixed distributions, regardless of their kurtosis signs, does not exist. The most prominent choices of the nonlinearity are the hyperbolic tangent and the cubic nonlinearity for super- and sub-Gaussian distributions, respectively. However, there are special distributions (with positive kurtosis) that are not separable by the “standard” hyperbolic tangent function. For complex signals we get the choice of using phase-preserving nonlinearities or individually applying the nonlinearities to real and imaginary parts of the signal. If real and imaginary parts of the signals are treated separately, phase equalization up to an ambiguity of $\pi/2$ can be achieved, a fact that might be exploited for phase synchronization. Three different options of algorithm modification are available for the case of instability of a given combination of nonlinearity and source distribution.

Chapter 5

The Threshold Nonlinearity

As has been shown in the last chapter, a single fixed nonlinearity for the blind separation of all non-Gaussian signals does not exist. In this chapter, a computationally simple nonlinearity in the form of a threshold function is introduced. Using this nonlinearity, convergence of BSS algorithms is shown to be robust, fast, and comparable to that of algorithms employing more complex polynomial nonlinearities. The general threshold nonlinearity (with an appropriate threshold) can separate any non-Gaussian, symmetrically distributed signals, as will be shown in theory and simulation.

After the formal derivation of the score function for the uniform distribution in Section 5.1, the stable region of the threshold parameter for some specific distributions is derived in Section 5.3, and optimal values for best separation performance are given. After some computer simulations in Section 5.4, Section 5.5 contains a proof that the threshold nonlinearity can separate any non-Gaussian signals. If the threshold parameter is made adaptive during the separation process, the successful separation of signals whose distribution is unknown is demonstrated and compared against other known methods in Section 5.6.

Finally, in Section 5.7 an extension to the standard algorithm mitigates separation coefficient bias, which is introduced by additive noise at the sensors. Some computer simulations support the theory.
5.1 Formal derivation of the score function for the uniform distribution

A very simple nonlinear function for the separation of sub-Gaussian signals is the monomial nonlinearity as given by Eq. (4.20). A monomial is the score function of a sub-Gaussian signal, a fact that is utilized in the following. Differentiating Eq. (2.15) with respect to \( u_i \) leads to

\[
p^j_i(u_i) = -\alpha \left( \frac{|u_i|}{\beta} \right)^{\alpha-1} \frac{\alpha}{\beta} e^{-\left( \frac{|u_i|^\alpha}{\beta} \right)}.
\]  
(5.1)

If we divide Eq. (5.1) by Eq. (2.15) and flip the sign, we get

\[
g_i(u_i) = -\frac{p^j_i(u_i)}{p_i(u_i)} = \alpha \left( \frac{|u_i|}{\beta} \right)^{\alpha-1} \frac{\alpha}{\beta} = \frac{\alpha}{\beta^\alpha} |u_i|^\alpha \frac{\alpha}{\beta} \text{sign}(u_i).
\]  
(5.2)

For unit variance, we can find \( \beta \) from the general expression for the \( n \)-th order moment of a generalized Gaussian signal, see Eq (2.29) or Table 2.1. For \( n = 2 \) and unit variance, Eq. (2.30) gives

\[
\beta = \frac{\Gamma \left( \frac{1}{2} \right)}{\Gamma \left( \frac{1}{2} \right)}.
\]  
(5.3)

Inserting this value for \( \beta \) into Eq. (5.2) yields the nonlinear function

\[
g_i(u_i) = \alpha \left( \frac{\Gamma \left( \frac{3}{2} \right)}{\Gamma \left( \frac{1}{2} \right)} \right)^{\alpha/2} \text{sign}(u_i) \cdot |u_i|^{\alpha-1}.
\]  
(5.4)

Eq. (5.4) is the score function for any generalized unit-variance Gaussian distribution. Using \( \Gamma(x) \cdot \Gamma(1-x) = \pi / \sin(\pi x) \) (see for example [21]) leads to

\[
g_i(u_i) = \alpha \left( \frac{\Gamma \left( \frac{3}{2} \right)}{\Gamma \left( \frac{1}{2} \right)} \right)^{\alpha/2} \frac{\Gamma \left( \frac{1}{2} \right)}{\Gamma \left( 1 - \frac{1}{2} \right)} \text{sign}(u_i) \cdot |u_i|^{\alpha-1}.
\]  
(5.5)

Both terms \( \Gamma \left( 1 - \frac{1}{2} \right) \) and \( \Gamma \left( 1 - \frac{3}{2} \right) \) are close to \( \Gamma(1) = 1 \) for large values of \( \alpha \), so that simplification of Eq. (5.5) yields

\[
g_i(u_i) \bigg|_{\alpha \gg 1} \approx \alpha \left( \frac{\sin \frac{\pi}{\alpha}}{\sin \frac{\pi}{\alpha}} \right)^{\alpha/2} \text{sign}(u_i) \cdot |u_i|^{\alpha-1}.
\]  
(5.6)

The first term of the Taylor series expansion of a sine function for a small argument is just the argument itself, leading to

\[
\left. g_i(u_i) \right|_{\alpha \gg 1} \approx \alpha \left( \frac{1}{3} \right)^{\alpha/2} \text{sign}(u_i) \cdot |u_i|^\alpha = a \frac{1}{u_i} \left( \frac{u_i^2}{3} \right)^{\alpha/2}.
\]  
(5.7)

We are now interested in the form of \( g_i(\cdot) \) as \( \alpha \) approaches infinity, in which case Eq. (2.15) corresponds to a uniform distribution. As a consequence of the behavior of \( \lim_{a \to \infty} a^b \) depending on \( |a| \) being smaller or greater than one, we can write the threshold nonlinearity as

\[
\lim_{\alpha \to \infty} g_i(u_i) = \begin{cases} -\infty, & u_i \leq -\sqrt{3}, \\ 0, & |u_i| < \sqrt{3}, \\ \infty, & u_i \geq \sqrt{3}. \end{cases}
\]  
(5.8)

The normalized uniform distribution only has a finite probability density for \( |u_i| < \sqrt{3} \); outside it is zero. With \( g_i(u_i) \) being zero for small \( u_i \), \( W_i + 1 \) in Eq. (3.17) grows gradually, thereby increasing \( u_i \). When \( u_i \) “hits” the threshold, it is pushed back hard (infinite gain) into the region where \( g_i(u_i) = 0 \), so that the amplitude of \( u_i \) is clearly controlled. The infinite gain Eq. (5.8) will of course cause convergence problems for a finite step size \( \mu \). The gain can therefore be traded off against a lower threshold \( \vartheta \) for a specified output power. Again, if we aim at unity output power, we need to scale the nonlinearity. Hence, for every component \( u_i \) of the vector \( u \), \( i = 1, \ldots, M \), we need to scale \( g_i(u_i) \) such that Eq. (4.8) is satisfied if \( p_i(\cdot) \) is a source distribution with unit variance \( \sigma_i^2 = 1 \). In fact, Eq. (4.8) is still valid in this form for source distributions \( p_i \) with arbitrary power \( \sigma_i^2 \). By satisfying Eq. (4.8), the output power of \( u \) will become that of the input

\[
E \left\{ u_i u_i^* \right\} = 1 \cdot \sigma_i^2.
\]  
(5.9)

Replacing Eq. (5.8) with

\[
g_i(u_i) = \begin{cases} 0, & |u_i| < \vartheta, \\ a \text{sign}(u_i), & |u_i| \geq \vartheta, \end{cases}
\]  
(5.10)
we get the gain $a$ of the threshold nonlinearity by solving Eq. (4.8) as

$$a = \frac{2\sqrt{3}}{3 - \vartheta^2} \quad (5.11)$$

for $0 \leq \vartheta < \sqrt{3}$. The resulting threshold nonlinearity is depicted in Fig. 5.1. It is a piece-wise constant function, and its form represents a three-level quantizer. Note that $a$ is always positive for the assigned range of $\vartheta$.

A similar nonlinearity in the form of a multiple-level quantizer has been introduced and optimized for the EASI algorithm by Kassam et al. [79].

**5.2 The threshold nonlinearity as a score function**

Since the threshold nonlinearity is a clipped version of the true score function for the uniform distribution, the question might be asked as to what distribution class Eq. (5.10) is the true score function for. The answer is given below, followed by Fig. 5.2, which illustrates these distributions parameterized by $\vartheta$.

The distribution is given by the pdf

$$p_x(x) = \begin{cases} \frac{1}{2(\vartheta + \frac{1}{2})} e^{-|x|/\vartheta}, & x < -\vartheta, \\ \frac{1}{2(\vartheta + \frac{1}{2})}, & -\vartheta \leq x \leq \vartheta, \\ \frac{1}{2(\vartheta + \frac{1}{2})} e^{-|x|-\vartheta}, & x > \vartheta. \end{cases} \quad (5.12)$$

For unit-variance signals the parameter $a$ becomes a dependent variable, with the relation

$$\frac{\vartheta^3}{3} - \vartheta = \frac{1}{a} - \frac{2}{a^3}. \quad (5.13)$$

**5.3 Stability analysis**

**5.3.1 Continuous distributions**

By experiment, a good value of the threshold parameter for uniform distributions was found to be $\vartheta = 1.5$. On the other hand, for the Laplacian distribution, which is an example of a super-Gaussian distribution and can be written in the form of Eq. (2.15) with $\alpha = 1$, Eq. (5.4) simplifies to

$$g_s(u_t) = \sqrt{2} \text{sign}(u_t). \quad (5.14)$$
which is the same as Eq. (5.10) for \( a = \sqrt{2} \) and \( \vartheta = 0 \). The signum function can hence be regarded as a threshold function with the threshold \( \vartheta \) set to zero.

In the following we use the stability analysis to assess the valid threshold values for some distributions. The source power is assumed to be normalized to one, i.e., \( E\{s^2\} = 1 \), and we scale the nonlinearity such that \( E\{g_i(s_i)\} = 1 \). Although the threshold function is not differentiable at \( s_i = \pm \vartheta \), we can derive \( E\{g_i(s_i)\} \) by the use of \( \delta \)-distributions and assuming a symmetric distribution

\[
E\{g_i(s_i)\} = \int_{-\infty}^{\infty} p_s(u_i) g_i(u_i) du_i \\
= \int_{-\infty}^{\infty} p_s(u_i) a(\delta(u_i + \vartheta) + \delta(u_i - \vartheta)) du_i \\
= 2a \cdot p_s(\vartheta).
\]  

Eqs. (3.83) and (3.84) can therefore be written as

\[
\kappa_+ = 2a \cdot p_s(\vartheta) + 1, \quad (5.16) \\
\kappa_- = 2a \cdot p_s(\vartheta) - 1. \quad (5.17)
\]

For a positive scaling factor \( a \), Eq. (5.16) is always positive. To make Eq. (5.17) positive, we must ensure that \( 2a \cdot p_s(\vartheta) > 1 \) by a suitable choice of \( \vartheta \). For the threshold nonlinearity and symmetric distributions, Eq. (4.8) can be written as

\[
2a \int_{\vartheta}^{\infty} p_s(u_i) u_i du_i = 1 \quad (5.18)
\]

or, if solved for the scaling factor,

\[
a = \frac{1}{2 \int_{\vartheta}^{\infty} p_s(u_i) u_i du_i}. \quad (5.19)
\]

Thus, the stability condition according to Eq. (3.85) for the threshold nonlinearity results in

\[
\frac{p_s(\vartheta)}{\int_{\vartheta}^{\infty} p_s(u_i) u_i du_i} > 1. \quad (5.20)
\]

Eq. (5.20) defines a stable region for \( \vartheta \) depending on the source distribution. In order to find the optimal values for \( \vartheta \) (in the sense of quality of separation), we have to minimize the term

\[
\frac{\xi}{\kappa_-} = \frac{\gamma_+/\kappa_+ + \gamma_-/\kappa_-}{2\kappa_-} \quad (5.21)
\]

as seen in Section 4.4, with \( \kappa_+, \kappa_-, \gamma_+, \) and \( \gamma_- \) defined by Eqs. (3.83), (3.84), (4.16), and (4.17), respectively. For stable solutions we assume \( \kappa_- > 0 \). For the threshold nonlinearity we can write

\[
E\{g_i^2(u_i)\} = 2a^2 \int_{\vartheta}^{\infty} p_s(u_i) du_i, \quad (5.22)
\]

We then get

\[
\gamma_+ = 2a^2 \int_{\vartheta}^{\infty} p_s(u_i) du_i + 1, \quad (5.23)
\]

\[
\gamma_- = 2a^2 \int_{\vartheta}^{\infty} p_s(u_i) du_i - 1, \quad (5.24)
\]

and thus

\[
\xi = \frac{\gamma_+ / \kappa_+ + \gamma_- / \kappa_-}{2} \\
= \frac{1}{2} \left( \frac{2a^2 \int_{\vartheta}^{\infty} p_s(u_i) du_i + 1}{2a p_s(\vartheta) + 1} + \frac{2a^2 \int_{\vartheta}^{\infty} p_s(u_i) du_i - 1}{2a p_s(\vartheta) - 1} \right) \\
= \frac{4a^3 p_s(\vartheta) \int_{\vartheta}^{\infty} p_s(u_i) du_i - 1}{4a^2 p_s^2(\vartheta) - 1} \quad (5.25)
\]

and

\[
\frac{\xi}{\kappa_-} = \frac{4a^3 p_s(\vartheta) \int_{\vartheta}^{\infty} p_s(u_i) du_i - 1}{(4a^2 p_s^2(\vartheta) - 1)(2a p_s(\vartheta) - 1) \kappa_-} \quad (5.26)
\]

so the optimal value for the threshold is

\[
\vartheta_{\text{opt}} = \arg\min_{\vartheta} \frac{\xi}{\kappa_-} \quad (5.27)
\]

\[
= \arg\min_{\vartheta} \frac{4a^3 p_s(\vartheta) \int_{\vartheta}^{\infty} p_s(u_i) du_i - 1}{(4a^2 p_s^2(\vartheta) - 1)(2a p_s(\vartheta) - 1)}. \quad (5.28)
\]

In Table 5.1 the performance parameters for the threshold nonlinearity are given. Fig. 5.3 shows the stability region and the optimal values of \( \vartheta \) of the
threshold nonlinearity for the generalized Gaussian distribution. Note that in this figure and in all further stability plots it is assumed that the gain is adjusted to result in unit-power output. Otherwise the stable areas are scaled, but the general shape remains. For signals with \(0 < \alpha \leq 2\), the stable range for \(\vartheta\) is roughly between zero and 0.5 with an optimal value of zero. For this threshold value, the threshold function is the true score function for the Laplace distribution. Interestingly, the optimal value for \(1 < \alpha < 2\) is slightly higher than zero. At the other extreme is the uniform distribution with \(\alpha = \infty\). While the upper limit of the stability range is \(\sqrt{3}\), the lower limit is one. The optimal value approaches \(\sqrt{3}\) as the pdf gets close to the uniform distribution. Note that only for the uniform distribution the threshold function is the score function (except for the finite gain). In practice, lower values of \(\vartheta\) lead to smaller gains of the nonlinearity, so for an adaptive system with a small step size, values of \(\vartheta\) lower than \(\sqrt{3}\) lead to reduced excess noise. Generally speaking, for sub-Gaussian signals, the stable range for \(\vartheta\) is between one and \(\sqrt{3}\). By decreasing \(\alpha\) we approach the normal distribution (\(\alpha = 2\)), which clearly poses a singularity in the stability plot. This can be seen by the broken stability regions at \(\alpha = 2\).

**Figure 5.3:** Stable range and optimal value of the threshold \(\vartheta\) for the generalized Gaussian as a function of the distribution parameter \(\alpha\).
In the following we take a closer look at two distributions, specifically the Laplacian and the uniform distribution as representatives of super- and sub-Gaussian distributions, respectively, and work out the exact optimal values for the threshold \( \theta \). For the Laplacian distribution, which is given by Eq. (2.15) with \( \sigma = 1 \) or, for unit variance,

\[
p_x(s) = \frac{1}{\sqrt{2}} e^{-\sqrt{2}|s|},
\]

we get with Eq. (5.19) for the gain of the threshold nonlinearity

\[
a = \frac{e^{\sqrt{2} \theta}}{\theta + \frac{\sqrt{2}}{2}}.
\]

Using Eq. (5.30) in Eq. (5.27) we find after some calculations the optimal threshold as

\[
\theta_{\text{opt}} = \arg\min_{\theta} \frac{\sqrt{2} e^{\sqrt{2} \theta} - \left( \theta + \frac{\sqrt{2}}{2} \right)^3}{\left( 2 - \left( \theta + \frac{\sqrt{2}}{2} \right)^2 \right) \left( \frac{\sqrt{2}}{2} - \theta \right)},
\]

which by a numerical inspection reveals \( \theta_{\text{opt}} = 0 \). The resulting nonlinearity is the much expected signum function. A similar calculation can be carried out for the uniform distribution, given by

\[
p_x(s) = \begin{cases} \frac{1}{2\sqrt{\pi}}, & |s| \leq \sqrt{3}, \\ 0, & \text{otherwise,} \end{cases}
\]

where the scaling factor \( a \) has already been computed by Eq. (5.11). Eq. (5.27) then results in

\[
\theta_{\text{opt}} = \arg\min_{\theta} \frac{8 \sqrt{3} - (3 - \theta)^2 (\sqrt{3} + \theta)}{4 - (3 - \theta)^2 (1 - \theta^2)}.
\]

A numerical inspection of Eq. (5.33) on the interval \([0, \sqrt{3}]\) shows that \( \theta_{\text{opt}} = \sqrt{3} \). These results are summarized in Table 5.2.

In the following we perform the stability analysis for two more families of distributions, the two-tailed Gamma and the zero-mean symmetric Beta distribution. The two-tailed Gamma family covers both sub- and super-Gaussian distributions, so it is unlikely to find a threshold value which results in a stable algorithm for all members of that family regardless of their parameter \( \alpha \). As can be seen in Fig. 5.4, the threshold value should be set close to zero for low \( \alpha \) and higher for larger \( \alpha \). For very high \( \alpha \), inferring a strong bimodality, the threshold is to be set to one. Another stability analysis for binary signals confirming this finding is following in the next section.

Zero-mean symmetric Beta distributed signals are always sub-Gaussian, so we expect—from the behavior of the threshold nonlinearity for the generalized Gaussian family—threshold values around one and higher. Such a behavior is confirmed by Fig. 5.5. As \( \alpha \) gets higher and the distribution approaches the Gaussian, the stable range for \( \theta \) increases, albeit the separation gets harder.

<table>
<thead>
<tr>
<th>pdf</th>
<th>( \theta_{\min} )</th>
<th>( \theta_{\max} )</th>
<th>( \theta_{\text{opt}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laplacian</td>
<td>0</td>
<td>( \sqrt{2}/2 )</td>
<td>0</td>
</tr>
<tr>
<td>uniform</td>
<td>1</td>
<td>( \sqrt{3} )</td>
<td>( \sqrt{3} )</td>
</tr>
</tbody>
</table>

Table 5.2: Range and optimal values of the threshold parameter for two special cases of generalized Gaussian distributed signals.

Figure 5.4: Stable range and optimal value of the threshold \( \theta \) for the two-tailed Gamma distribution as a function of the distribution parameter \( \alpha \).
5.3.2 Discrete distributions

For discrete distributions, where the pdf is rather a pmf (probability mass function), the stability condition of both the sign function and the threshold nonlinearity is not satisfied if there is no mass at the discontinuity of the nonlinearity. If, however, by fluctuations of the adaptation, residual mixing produces a finite density at these points, the algorithm can stabilize.

In the following, the stability analysis of the last section is extended to discrete distributions. We start again with the stability condition for the threshold nonlinearity according to Eq. (5.20). Whereas the integral in the denominator of Eq. (5.20) can be written as a sum for discrete distributions,

\[ \int_{-\infty}^{\infty} p_{u}(u) u du = \sum_{k, A_k \geq \theta} \Pr(u_i = A_k) A_k, \quad (5.34) \]

the evaluation of the probability density, as appearing in the numerator of Eq. (5.20), needs a closer look. Close to an equilibrium point we may model the output distribution as a convolution of the discrete probability model of the sources by some mixing noise distribution, which is Gaussian distributed. This is well justified for a large number of sources or in communication systems by additive noise present. In other words, the discrete-level distribution is convolved with the probability density function (pdf) of a Gaussian noise signal. Thus, the probability density at a certain constellation point is therefore the discrete probability of that point multiplied (as a result of the convolution with a Dirac impulse) by the mode of the Gaussian kernel

\[ \frac{1}{\sqrt{2\pi \sigma_n^2}} \]

with \( \sigma_n^2 \) being the variance of the mixing noise. The resulting pdf for a 4-PAM (Pulse Amplitude Modulation) signal with an SNR = 25 dB is depicted in Fig. 5.6. The stability regions are thus dependent on the mixing noise. Figs. 5.7 and 5.8 show the stable regions as derived by the evaluation of Eq. (5.20) for BPSK (Binary Phase Shift Keying) and 4-PAM, respectively. It is interesting to note that in addition to the region around the outer symbols, which looks similar for BPSK and 4-PAM, there is a further stable region around the inner symbols in the case of 4-PAM.
5.3. Stability analysis

It becomes apparent that for a stable update equation for BPSK signals for a fixed scaling constant, the threshold $\theta$ has to be in the neighborhood of the symbol amplitude, otherwise the algorithm becomes unstable. A closer look at Eq. (5.20) reveals that the mixing noise keeps the algorithm stable through a finite pdf in the neighborhood of the symbol amplitude. In other words, if the threshold $\theta$ is chosen too far away from the symbol amplitude, more mixing noise is needed to satisfy Eq. (5.20). For BPSK, the threshold $\theta$ should therefore be chosen directly at the symbol amplitude $A_1 = 1$. For this choice, with probability 0.5 the signal will be larger (smaller) than the threshold, enforcing a choice of the scaling factor $a = 2$ in order to satisfy the scaling condition in Eq. (4.8). For all choices of the threshold $\theta$ smaller than $A_1 = 1$ and low residual mixing, a scaling factor of $a = 1$ is needed. For larger threshold values, the gain gets impractically high due to Eq. (4.8). For $M$-PAM signals with $M > 2$, stable algorithms can be obtained by setting the threshold to the outermost symbol amplitude (cf. Table B.1)

$$\theta = \sqrt{\frac{3(M-1)}{M+1}}.$$  \hspace{1cm} (5.35)

The corresponding gain is

$$a = M \sqrt{\frac{M+1}{3(M-1)}}.$$  \hspace{1cm} (5.36)

5.3.3 Complex-valued baseband signals

Similar stability analyses to the ones shown in the last section can be carried out for complex-valued baseband signals with discrete probability distributions. As a general rule, we find that placing the threshold value to the outermost symbol amplitude produces stable algorithms. The amplitude is the magnitude of the complex signal if applying phase-preserving algorithms and the real (imaginary) part for I/Q-independent nonlinearities, respectively. Similarly to Eqs. (5.35) and (5.36) we can find the threshold and gain values for the phase-preserving threshold nonlinearity from Table B.1 as

$$\theta = \sqrt{\frac{3\sqrt{M-1}}{\sqrt{M+1}}}.$$  \hspace{1cm} (5.37)

$$a = \frac{M}{2} \sqrt{\frac{\sqrt{M+1}}{3(\sqrt{M}-1)}}.$$  \hspace{1cm} (5.38)
and for I/Q-independent nonlinearities as

\[ \vartheta = \sqrt{\frac{3(\sqrt{M} - 1)}{2(\sqrt{M} + 1)}}, \]  
\[ a = \sqrt{\frac{M(\sqrt{M} + 1)}{6(\sqrt{M} - 1)}}. \]  

5.4 Computer simulations

5.4.1 Uniformly distributed signals

To separate uniformly distributed real-valued signals, very often a nonlinearity of the form given by Eq. (4.20) is used. Such nonlinearities shall be compared with the threshold nonlinearity using numerical computer simulations. For the following simulation of the convergence behavior of blind signal separation using the threshold function, \( M_d = 10 \) independent, uniformly distributed source signals are mixed by the matrix \( A \), whose condition number is chosen \( \chi(A) = 100 \) (the singular values of \( A \) are logarithmically distributed). The step size \( \mu \) is tuned such as to reach a residual mixing of \( J_{\text{CM}}(P) = -20 \text{dB} \). Fig. 5.9 displays the performance curves for different nonlinearities, such as Eq. (4.20) for different \( p \). Clearly, the threshold function shows a convergence behavior comparable to that of more complicated nonlinearities. Best results were achieved with a threshold value of \( \vartheta = 1.5 \).

5.4.2 M-QAM signals

The phase-preserving threshold nonlinearity for QAM signals is given by

\[ g_i(u_i) = \begin{cases} 0, & |u_i| < \vartheta, \\ \frac{u_i}{|u_i|}, & |u_i| \geq \vartheta. \end{cases} \]

For 16-QAM distributions, the values according to Eqs. (5.37) and (5.38) are \( \vartheta = 1.34 \) and \( a = 5.96 \). On the other hand, if the threshold nonlinearity is applied to the real and imaginary parts separately, we get from Eqs. (5.39) and (5.40) \( \vartheta = 0.95 \) and \( a = 2.11 \).

Similarly to the last experiment for uniformly distributed signals, ten 16-QAM signals were mixed using a complex mixing matrix \( A \) with \( \chi(A) = 100 \) and logarithmically spaced singular values. Performance curves are depicted in Fig. 5.10. It can be observed that the cubic nonlinearity applied individually to the real and imaginary parts separates the signals fastest. For the other nonlinearities, the differences are small. Fig. 5.10 reveals an interesting behavior to the algorithm: Although the initial slopes are quite different, after the algorithm have all reached a point of achieving the right scaling of the coefficients, the slopes become all equally steep. In terms of complexity, individually applying the threshold nonlinearity to the real and imaginary parts of the signal is advantageous, since no absolute value needs to be computed. This works of course only if no carrier frequency offset occurs, i.e., the frequency offset needs to be addressed prior to this algorithm, which in turn takes care of phase offsets. The effect on the back-rotation of the signal constellation is shown in Fig. 5.11.

For simulation results of other constellations see also [100].
5.5 The threshold nonlinearity as a universal parametric nonlinearity

The threshold nonlinearity [100] with $\vartheta = A_1$ and $\alpha = 2$ successfully separates the “challenging” distribution given in Section 4.7.4, which was verified both by inspection of the stability condition as well as experimental simulation. Fig. 5.12 shows the convergence performance of different nonlinearities for ten sources with a “challenging” distribution according to Section 4.7.4. All but the threshold nonlinearity fail to separate the signals. This leads to the question whether the threshold nonlinearity is capable of separating any non-Gaussian distribution for an appropriate threshold parameter $\vartheta$, thus satisfying Eq. (5.20). The answer is given by the following theorem. In contrast to Eq. (5.10) we omit scaling but reintroduce it in a subsequent step.

Figure 5.10: Convergence curves of blind separation with different nonlinearities for 16-QAM signals.

Figure 5.11: Signal constellation diagrams of 16-QAM signals after separation with different nonlinearities.

Figure 5.12: Convergence of different nonlinearities for a mixture of signals with the “challenging” distribution.
Theorem 5.1

The threshold nonlinearity given by

\[ g(u) = \begin{cases} 0, & |u| < \theta, \\ \text{sign}(u), & |u| \geq \theta \end{cases} \]

satisfies the local stability condition

\[ \sigma_u^2 p_u(\theta) - \int_\theta^\infty p_u(u) u \, du > 0, \quad (5.43) \]

for some appropriately chosen threshold value \( \theta \geq 0 \) and any continuous, differentiable, non-Gaussian output distribution \( p_u(\cdot) \). In addition we have

\[ \sigma_u^2 p_u(\theta) - \int_\theta^\infty p_u(u) u \, du = 0, \quad \forall \theta \in \mathbb{R}^+_0, \quad (5.44) \]

if and only if \( p_u(\cdot) \) is Gaussian.

The proof of Theorem 5.1 is one of existence rather than of construction in that it shows that there is a threshold parameter \( \theta \) for which the update equation Eq. (3.17) is stable, but it does not necessarily give an explicit solution for \( \theta \).

Proof: We consider real, symmetric, continuous, differentiable distributions. The result for other distributions can be obtained by convolving discrete distributions by low-variance Gaussian kernels. We have to show that to satisfy the stability condition Eq. (3.85), the inequality

\[ \sigma_u^2 p_u(\theta) - \int_\theta^\infty p_u(u) u \, du > 0 \]

has to be satisfied for at least one value of \( \theta \in \mathbb{R}^+_0 \), given a non-Gaussian distribution. We assume that no value of \( \theta \) can satisfy Eq. (5.45), so

\[ \sigma_u^2 p_u(\theta) \leq \int_\theta^\infty p_u(u) u \, du, \quad \forall \theta \in \mathbb{R}^+_0, \quad (5.46) \]

and lead the proof by contradiction. First we show that for a normal distribution \( p_u(\cdot) = \mathcal{N}(0, \sigma_u^2) \), we have

\[ \sigma_u^2 p_u(\theta) \equiv \int_\theta^\infty p_u(u) u \, du, \quad \forall \theta \in \mathbb{R}^+_0, \quad (5.47) \]

5.5. The threshold nonlinearity as a universal parametric nonlinearity

To this end, we assume that

\[ \sigma_u^2 p_u(\theta) - \int_\theta^\infty p_u(u) u \, du = c \]

for some nonpositive constant \( c \). Taking derivatives of both sides of Eq. (5.48) with respect to \( \theta \) gives the differential equation

\[ \sigma_u^2 \frac{d p_u(\theta)}{d \theta} + \theta p_u(\theta) = 0. \quad (5.49) \]

Eq. (5.49) is a simple first-order differential equation whose parametric solution is

\[ p_u(\theta) = K \exp \left( -\frac{\theta^2}{2\sigma_u^2} \right), \quad K \geq 0. \quad (5.50) \]

Because \( p_u(\cdot) \) is a pdf, the value of \( K \) must be \( K = 1/(\sqrt{2\pi\sigma_u}) \), meaning that \( c = 0 \). This proves the uniqueness of the Gaussian distribution as the pdf that minimizes the LHS of the stability condition inequality. All other continuously valued and differentiable distributions should therefore satisfy the inequality.

In the following we make use of crossing points of the different pdfs, which are given as points where these two functions are identical. By taking \( \theta \) as the last (rightmost) crossing point of the distribution under consideration and the normal distribution, see Fig. 5.13, we have either Eq. (5.45), which is already in contradiction to Eq. (5.46), or

\[ \sigma_u^2 p_u(\theta) < \int_\theta^\infty p_u(u) u \, du \quad (5.51) \]

for some region around that particular \( \theta \). By integrating both sides of Eq. (5.46) over \( \mathbb{R}^+_0 \), we get

\[ \sigma_u^2 \int_\theta^\infty p_u(\theta) d\theta = \sigma_u^2 \int_\theta^\infty \int_0^\infty p_u(u) u \, du \, d\theta, \quad (5.52) \]

where the strict inequality results from the region where Eq. (5.51) is valid. The right-hand side of Eq. (5.52) can be solved by exchanging the integrals

\[ \int_0^\infty \int_\theta^\infty p_u(\theta) u \, du \, d\theta = \int_0^\infty \int_0^\infty d\theta \, p_u(u) u \, du = \int_0^\infty p_u(u) u^2 \, du = \frac{\sigma_u^2}{2}. \quad (5.53) \]
Eq. (5.53) is a contradiction to Eq. (5.52). This means that if there are values of \( \vartheta \) satisfying Eq. (5.51), due to Eq. (5.53) there must also be values satisfying Eq. (5.54) and vice versa, which is in contradiction to Eq. (5.46).

Note that we have not used the notion of sub- and super-Gaussian to refer to the tail of the distribution, since also multimodal distributions are considered, where the tail observation is still valid, but the use of the terms sub- and super-Gaussian is not appropriate.

From Theorem 5.1, we immediately get the following corollary.

**Corollary 5.1**

The threshold nonlinearity given by

\[
g(u) = \begin{cases} 
0, & |u| < \vartheta, \\
a \text{sign}(u), & |u| \geq \vartheta
\end{cases}
\]

satisfies the local stability condition for some appropriately chosen threshold value \( \vartheta \geq 0 \) and any continuous, differentiable, non-Gaussian output distribution \( p_u(.) \).

**Proof:** The proof of Theorem 5.1 can easily be adapted by taking into account an arbitrary scaling factor \( a \).

---

5.6 Stabilization of arbitrary distributions

In contrast to Theorem 5.1 we have used here a more general threshold nonlinearity with a gain parameter \( a \). As in previous examples, this additional degree of freedom allows us to achieve a desired output variance without going through rescaling.

5.6.1 Known methods

In practice, the distributions of the source signals are often identical, since the nature of their origin is related. In this case, the activation function will be the same for all output nodes. However, we may find the situation where some source signals have different distributions, possibly with a different sign of their respective kurtoses. If in a mixture some source signals are sub-Gaussian (negative kurtosis) and some super-Gaussian (positive kurtosis) distributed, the appropriate activation function might be chosen in advance, as long as the number of sub- and super-Gaussian sources is known. Doing so, the neural network is deprived of some degree of freedom, due to the restriction of permutation within the group of equal kurtosis sign. In other words, once an activation
function is chosen, only a signal with the appropriate kurtosis sign can be separated at that specific output node. Other signals are forced away to output nodes with the appropriate activation function. The learning speed can be greatly accelerated by letting the network choose its permutation closest to some initial mixing condition. This can be achieved by an adaptive activation function. If the number of sub-Gaussian and the number of super-Gaussian sources is unknown, adaptive activation functions are a necessity.

Douglas et al. [48] switch between two nonlinearities, namely
\[ g_N(u_i) = u_i^3 \quad \text{and} \quad g_P(u_i) = \tanh(10u_i), \] (5.55)
where \( g_N(\cdot) \) and \( g_P(\cdot) \) separate sub- and super-Gaussian signals, respectively. The algorithm does not try to normalize its output power regardless of the distribution. The stability condition [131] therefore includes information on the variances, and can be written as
\[ \frac{E\{g'_N(u_i)u_i^2\}}{E\{g_N(u_i)u_i\}} - \frac{E\{g'_P(u_i)u_i^2\}}{E\{g_P(u_i)u_i\}} > 1 \] (5.56)
for any two outputs \( i \neq j \). A stronger and thus sufficient condition for Eq. (5.56) to hold is
\[ \frac{E\{g'_N(u_i)u_i^2\}}{E\{g_N(u_i)u_i\}} > 1 \] (5.57)
for each output \( i \). A sufficient stability condition for the nonlinearities in Eq. (5.55) is therefore
\[ E\{g'_N(u_i)u_i^2\} - E\{g_N(u_i)u_i\} > 0. \] (5.58)
In their adaptive algorithm, the left-hand side of Eq. (5.58) is constantly evaluated for the two nonlinearities \( g_N(\cdot) \) and \( g_P(\cdot) \). The larger value decides which activation function is being applied. A variant of this technique has been come up by Ihm et al. [70]. Their algorithm performs a soft change-over from one nonlinearity to the other based on the sample kurtosis.

Similarly, Lee et al. [87] present an extended Infomax algorithm, where the update equation for the separation matrix is formulated as
\[ W_{t+1} = W_t + \mu \left( I - K \tanh(\mu)u^T - uu^T \right) W_t, \] (5.59)
with \( K = \text{diag}[k_1, \ldots, k_M]^T \) being a diagonal matrix of signs. \( k_i \) is positive for a super-Gaussian and negative for a sub-Gaussian signal, respectively. The activation function can then be written as
\[ g_i(u_i) = k_i \tanh(u_i) + u_i. \] (5.60)
If the distributions are unknown, the sign might be switched according to a kurtosis estimation at the output node or some parameter expressing the stability of the nonlinearity currently used as the activation function. Similarly to Eq. (5.58) it follows
\[ k_i \left( 1 - E \left\{ \tanh^2(u_i) \right\} \right) E\{u_i^2\} - E \left\{ \tanh(u_i)u_i \right\} > 0. \] (5.61)
By choosing \( k_i \) the same sign as the rest of Eq. (5.61), the algorithm is stabilized. Thus, the sign \( k_i \) must be learned as we go along,
\[ k_i = \text{sign} \left( \left( 1 - E \left\{ \tanh^2(u_i) \right\} \right) E\{u_i^2\} - E \left\{ \tanh(u_i)u_i \right\} \right) \] (5.62)
or, in more compact form by using the substitution \( \text{sech}^2(\cdot) = 1 - \tanh^2(\cdot) \),
\[ k_i = \text{sign} \left( E \left\{ \text{sech}^2(u_i) \right\} \right) E\{u_i^2\} - E \left\{ \tanh(u_i)u_i \right\} \] (5.63)
Again, output powers are not normalized, and depend on the source distributions.

Yet another method was presented by Cichocki et al. [34], where basically two nonlinearities \( f(\cdot) \) and \( g(\cdot) \) are used in the update expression
\[ W_{t+1} = W_t + \mu \left( I - f(u)g(u^T) \right) W_t \] (5.64)
according to
\[ f_i(u_i) = \begin{cases} \tanh(10u_i), & k_4 > 0.1, \\ u_i, & k_4 < -0.1, \end{cases} \] (5.65)
\[ g_i(u_i) = \begin{cases} \tanh(10u_i), & k_4 < -0.1, \\ u_i, & k_4 > 0.1, \end{cases} \] (5.66)
where \( \hat{k}_4 \) is the sample kurtosis of the \( i \)th signal. A disadvantage of this method is that it fails for signals whose kurtosis is less than 0.1 in magnitude.
5.6.2 The adaptive threshold nonlinearity

Since we know that any generalized Gaussian distribution with $\alpha \neq 2$ can be separated by the threshold function with either $\theta = 0$ or $\theta \approx 1.5$, we can set up a neural network in which the update equation for the separation matrix is given by Eq. (3.17) with

$$g_i(u_i) = \begin{cases} 0, & |u_i| < \theta_i, \\ a_i \text{sign}(u_i), & |u_i| \geq \theta_i. \end{cases}$$

(5.67)

Each threshold $\theta_i$ is chosen from $[0, 1.5]$ as that value which maximizes the right-hand side of Eq. (3.84) with $g_i(\cdot)$ of Eq. (5.67). The use of the stability equation to switch between the two threshold values has two important disadvantages. First, the value of $p_i(\theta)$ is difficult to work out since the function $p_i(\cdot)$ is unknown. Second, although the threshold function successfully separates discrete distributions, $p_i(\theta)$ is generally zero for discrete distributions as used in data communications, making the switching criterion invalid.

A better alternative is to drive the threshold vector $\mathbf{\theta} \triangleq [\theta_1, \ldots, \theta_M]^T$ according to

$$\theta_{i+1} = \theta_i - \mu_\theta \mathbf{\kappa}_i,$$

(5.68)

where $\mu_\theta$ is a properly chosen positive constant, and $\mathbf{\kappa}_i \triangleq [\kappa_{1,i}, \ldots, \kappa_{M,i}]^T$ is an estimate of the output kurtoses of the vector $\mathbf{u}$ at sample time $t$, with

$$\kappa_{i,t} = \frac{L}{L-1} \sum_{\tau=0}^{L-1} u_{i,t-\tau}^4 - 3.$$  

(5.69)

Additionally, the elements of $\mathbf{\theta}_{i+1}$ are clipped at zero and 1.5 to keep them inside a meaningful region. In other words, a decision is made as to which threshold value leads to a more stable solution.

5.6.3 Output normalization

In the analysis of the stability we have shown that the scaling factors in the vector $\mathbf{\alpha} \triangleq [\alpha_1, \ldots, \alpha_M]^T$ have to be chosen according to Eq. (5.18) in order to obtain output signals with unit variance. In an environment where the probability distributions are given, Eq. (5.18) can be evaluated off-line and $\mathbf{\alpha}$ is thus fixed during learning and separation. If the distributions are unknown, however, $\mathbf{\alpha}$ itself has to be found during the learning mode. To this end, we note that for unimodal, symmetric distributions, $f_i$ is a monotonic decreasing function of the standard deviation $\sigma_i$ of the $i$th output. Vice versa, $\sigma_i$ can be written as $\sigma_i = f_i(\alpha_i)$, where $f_i(\alpha_i)$ is a monotonic decreasing function for $\alpha_i > 0$, hence

$$\frac{\partial f_i(\alpha_i)}{\partial \alpha_i} < 0.$$  

(5.70)

The exact course of $f_i(\cdot)$ depends on the pdf of the $i$th source. For convenience we denote $\mathbf{f}(\cdot) \triangleq [f_1(\cdot), \ldots, f_M(\cdot)]^T$. We define our error function $\mathbf{e} \triangleq [e_1, \ldots, e_M]^T$ by the deviation from unit variance

$$e_i = 1 - \hat{\sigma}_i^2.$$  

(5.71)

and its sum of squares as the cost function

$$J(\mathbf{\alpha}) \triangleq \mathbf{e}^T \mathbf{e} = \sum_{i=1}^{M} e_i^2 = \sum_{i=1}^{M} (1 - \hat{\sigma}_i^2)^2.$$  

(5.72)

$\hat{\sigma}_i^2$ denotes the estimation of the output power $\sigma_i^2$. The derivative of the cost function $J(\mathbf{\alpha})$ with respect to the gain $\mathbf{\alpha}$ is

$$\nabla_{\mathbf{\alpha}} J(\mathbf{\alpha}) = 2\mathbf{e} \odot \left[ \frac{\partial f_1(\alpha_1)}{\partial \alpha_1}, \ldots, \frac{\partial f_M(\alpha_M)}{\partial \alpha_M} \right]^T$$

$$= -4 \left[ (1 - \hat{\sigma}_1^2) \frac{\partial f_1(\alpha_1)}{\partial \alpha_1}, \ldots, (1 - \hat{\sigma}_M^2) \frac{\partial f_M(\alpha_M)}{\partial \alpha_M} \right]^T.$$  

(5.73)

$\odot$ denotes the element-wise multiplication of two vectors. We can now develop a stochastic-gradient algorithm to train the gain vector

$$a_{t+1} = a_t - \mu_\alpha \nabla_{\mathbf{\alpha}} J(\mathbf{\alpha}).$$  

(5.74)

Using Eq. (5.70) and the fact that $\hat{\sigma}_i > 0$, we can incorporate $\hat{\sigma}_i$ and $\frac{\partial f_i(\alpha_i)}{\partial \alpha_i}$ into a different step size $\mu_\alpha$ and write

$$a_{t+1} = a_t - \mu_\alpha (1 - \hat{\sigma}_i^2),$$  

(5.75)

with $\mathbf{1}$ being a vector of ones and $\hat{\sigma}_i^2 \triangleq [\hat{\sigma}_1^2, \ldots, \hat{\sigma}_M^2]^T$ the vector of power estimates, given by

$$\hat{\sigma}_{i,t}^2 = \frac{1}{L} \sum_{\tau=0}^{L-1} u_{i,t-\tau}^2.$$  

(5.76)
respectively. Eq. (5.75) is a simple AGC (automatic gain control) algorithm, which normalizes the output powers of the separation process. It runs along with the training of $W$ and $\theta$. Alternatively, the normalization can of course be performed by a separate AGC stage after the separation process. This is, for example, necessary if the mixture contains binary sources. It is straightforward to see that a normalized source with symbol values $\pm 1$ produces zero output after the threshold function with $\vartheta = 1.5$.

The structure of the resulting neural network is depicted in Fig. 5.14. Note that the training of the nonlinearity involves deciding on the thresholds as well as adapting the scaling factors. Similar ideas were used in an LMS-based algorithm in [51].

![Neural network with adaptive threshold nonlinearities](image)

**Figure 5.14:** Neural network with adaptive threshold nonlinearities.

In summary, the adaptive threshold nonlinearity algorithm is given on the following page:

---

### Adaptive Threshold Nonlinearity Algorithm

#### Initialization:

- $W_{t=0} = I$,  
  \[ (5.77) \]
- $\theta_{t=0} = 0.75 \cdot I$,  
  \[ (5.78) \]
- $a_{t=0} = 2 \cdot I$.  
  \[ (5.79) \]

#### Separation:

- $u_t = W_t x_t$.  
  \[ (5.80) \]

#### Weights learning:

- $W_{t+1} = W_t + \mu \left( I - g_i(u_i)u_i^T \right) W_t$,  
  \[ (5.81) \]
  with
  \[ g_i(u_i) = \begin{cases} 0, & |u_i| < \vartheta_i, \\ a_i \text{sign}(u_i), & |u_i| \geq \vartheta_i. \end{cases} \]  
  \[ (5.82) \]

#### Statistics estimation:

- $\hat{\sigma}_{i,t}^2 = \frac{1}{L} \sum_{\tau=0}^{L-1} u_{i,t-\tau}^2$,  
  \[ (5.83) \]
- $\hat{\kappa}_{i,t} = \frac{4}{\hat{\sigma}_{i,t}^4} - 3$.  
  \[ (5.84) \]

#### Nonlinearity update:

- $\vartheta_{i,t+1} = \begin{cases} 0, & \vartheta_{i,t} - \mu \varrho \hat{\kappa}_{i,t} < 0, \\ 1.5, & \vartheta_{i,t} - \mu \varrho \hat{\kappa}_{i,t} > 1.5, \\ \vartheta_{i,t} - \mu \varrho \hat{\kappa}_{i,t}, & \text{otherwise}, \end{cases}$  
  \[ (5.85) \]
- $a_{i+1} = a_t - \mu a_t (1 - \hat{\sigma}_{i,t}^2)$.  
  \[ (5.86) \]
5.6.4 Advantages of the new method

The adaptive threshold nonlinearity offers three main advantages over existing methods. First, it is not only easy to implement on a DSP, but would also be considerably simpler to implement in hardware as compared to polynomial nonlinearities or hyperbolic functions used in other methods, which either need look-up tables or polynomial approximations, since the set of possible output values of the threshold function only contains three values, ±a and 0. The threshold operation can thus be easily implemented by two comparators only. Second, the threshold nonlinearity can be stabilized for distributions which are impossible to separate by either \( g(u) = u^3 \) or \( g(u) = \tanh(u) \), see also [44]. Third, normalization of the output signals is achieved independently of the distributions, which is not the case with any of the existing methods in [34,48,87].

5.6.5 Computer simulations

For the following simulations of the learning time analysis of blind signal separation using a single-layer neural network with an adaptive threshold function, \( M_s = 10 \) independent source signals were mixed by a random matrix \( A \), whose condition number is chosen \( \chi(A) = 100 \) (the singular values of \( A \) are logarithmically distributed). Block processing with a block length \( L = 64 \) was applied. With this length the kurtosis estimation for the purpose of threshold learning is accurate enough, and inter-block memory does not offer any advantage.

In the first computer experiment we mixed three Laplacian, three uniform, three 16-PAM, and one Gaussian source. If more source signals are Gaussian distributed, they can still be separated from other sources by the adaptive threshold nonlinearity, but remain mixed among themselves, leading to a disturbed permutation matrix. This is an inherent limitation of blind separation using higher-order statistics, and is usually circumvented by the restriction to at most one Gaussian source. A neural network with the adaptive threshold nonlinearity algorithm, Eqs. (5.77) to (5.86), was then used to separate the signals in a block-processing manner. The step size \( \mu \) of the training was adjusted for a residual mixing of \( J_{	ext{cl}}(P) = -20 \text{dB} \), where the performance measure \( J_{	ext{cl}}(P) \) is the average interchannel interference given by Eq. (3.4). Practical values for the different step sizes were \( \mu = 0.0017 \), \( \mu_\theta = 0.32 \), and \( \mu_a = 0.05 \), respectively.

Fig. 5.15 shows the learning process. The effect of the AGC can be observed as well as the convergence of the kurtoses of the output signals to the respective values \( 3.0 \) for Laplacian, \( -1.2 \) for uniform, \( -1.209 \) for 16-PAM, and zero for Gaussian distributions. The threshold values approach either zero or 1.5, depending on their kurtoses, and converge around 10000 samples, except for the output node with the Gaussian distribution, where the threshold value remains undecided.

In the next simulation we wanted to compare the adaptive threshold nonlinearity algorithm with the algorithms found by Douglas et al. [48], Lee et al. [87], and Cichocki et al. [34]. To this end, we mixed five Laplacian and five uniform source signals. The four algorithms were then run with as similar parameters as possible to allow a fair comparison. A block processing with the block size \( L = 64 \) was used for all algorithms. The step sizes of the training algorithms were adjusted individually for a residual mixing of \( J_{	ext{cl}}(P) = -20 \text{dB} \). Averaging over 50 runs with different matrices (all with the characteristics as described above) were carried out to get typical behavior. Fig. 5.16 shows the
separation performance for all tested algorithms. The adaptive threshold non-
linearity algorithm, the algorithm by Douglas et al. and that by Cichocki et al.
reach the $-20$dB point at exactly the same time on average, whereas the
extended Infomax algorithm needs considerably more time.

![Learning curves for blind signal separation algorithm of mixed distributions.](image)

Figure 5.16: Learning curves for blind signal separation algorithm of mixed
distributions.

A comparison with simulations of Laplacian sources only shows that the
learning time associated with Laplacian sources only is about the same as the
learning time obtained here [73]. Simulations also confirm that the adaptive
threshold concept is advantageous for mixed-kurtosis signals even if the dis-
tributions are known, since by fixing the nonlinear function in advance we
reduce the degree of freedom of the neural network by restricting the distri-
butions to the output nodes with the appropriate nonlinear function, whereas
with the adaptive threshold, the network is free to choose among more per-
mutations, thus reducing learning time. As an example, for five sub- and five
super-Gaussian sources, the total degree of freedom is

$$
\#P = 10! \cdot 2^{10} = 3.7 \times 10^9, \tag{5.87}
$$

whereas the restricted degree of freedom is “only”

$$
\#P = 5! \cdot 5! \cdot 2^{10} = 1.5 \cdot 10^7. \tag{5.88}
$$

### 5.7 Separating in an AWGN channel

#### 5.7.1 Unbiased blind separation

Algorithms of the form given by Eq. (3.17) lead to a biased solution if additive
noise is present at the sensors. Particularly in communication environments we
often have this situation of additive noise. By linearly combining the signals
in order to separate them, the noise signals get correlated at the output, intro-
ducing dependencies between the sensor signals. Any criterion that searches
for the minimum dependence among the output signals will therefore deviate
from this solution, thereby introduce a bias. A combined learning process in-
volving unsupervised learning for the separation and supervised learning for
noise reduction was presented in [33]. The lack of a noise reference in prac-
tice, however, makes this approach inapplicable to most common problems.
It is possible to devise an update equation with an additional term in the update
equation, which involves either some expectation of the signal derivatives [32]
or their stochastic versions [49]

$$
W_{t+1} = W_t + \mu \left( I - g(u)u^T + BW_t R_n W_t^T \right) W_t, \tag{5.89}
$$

where $B$ is a diagonal matrix with entries

$$
b_{ii} = E \left\{ \frac{dg(u_t)}{du_t} \right\}, \tag{5.90}
$$

and $R_n$ is the covariance matrix of the noise contribution. To see the mechanism
behind Eq. (5.89), we define an unbiased estimate of the source signal as

$$
\hat{u} = W A s. \tag{5.91}
$$

If we use the original algorithm, Eq. (3.17), to separate a noisy mixture of
signals, we get an equilibrium point when the expectation of the parenthesis is
zero, hence

$$
E \left\{ I - g(u)u^T \right\} = 0. \tag{5.92}
$$

But since the output is noisy, i.e.,

$$
u = \hat{u} + Wn, \tag{5.93}
$$

we get from Eq. (5.92)

$$
E \left\{ I - g(\hat{u} + Wn)(\hat{u} + Wn)^T \right\} = 0. \tag{5.94}
$$
A first-order truncated Taylor series expansion of the nonlinearity around \( \hat{u} \) yields
\[
g(\hat{u} + Wn) = g(\hat{u}) + \text{diag}(g'(\hat{u})) Wn, \tag{5.95}
\]
where \( \text{diag}(g'(\hat{u})) \) is a diagonal matrix with the elements \( g'(\hat{u}) \) located on the diagonal. Inserted into Eq. (5.94) this results in
\[
E \left[ \mathbf{I} - g(\hat{u} + Wn)(\hat{u} + Wn)^T \right] = \mathbf{I} - E \left[ g(\hat{u}) \hat{u}^T \right] - E \left[ \text{diag}(g'(\hat{u})) Wn \hat{u}^T \right] = 0. \tag{5.96}
\]
Since the noiseless estimate is uncorrelated to the noise, the third and the fourth term of the RHS of Eq. (5.96) are zero, hence
\[
E \left[ \mathbf{I} - g(\hat{u} + Wn)(\hat{u} + Wn)^T \right] = \mathbf{I} - E \left[ g(\hat{u}) \hat{u}^T \right] = 0. \tag{5.97}
\]
The equilibrium point is therefore the point where the above equation is satisfied, and not the point at which the unbiased estimate of the source signals \( \hat{u} \) are independent. The third term of the RHS of Eq. (5.97) is now identified as the bias term and has to be subtracted in the original update equation, leading to Eq. (5.89). If we then make the same analysis on Eq. (5.89), for which we know that at the equilibrium we have
\[
E \left[ \mathbf{I} - g(\mathbf{u}) \mathbf{u}^T + B W_t R_n W_t^T \right] = 0, \tag{5.98}
\]
we get
\[
E \left[ \mathbf{I} - g(\mathbf{u}) \mathbf{u}^T + B W_t R_n W_t^T \right] = \mathbf{I} - E \left[ g(\hat{u}) \hat{u}^T \right] - E \left[ \text{diag}(g'(\hat{u})) W R_n W_t^T \right] = 0. \tag{5.99}
\]
Hence, the equilibrium means that the elements of \( \hat{u} \) will be mutually independent.

Although the threshold nonlinearity is nondifferentiable, its expectation can be expressed by integration over a Dirac impulse
\[
E \left[ g'(\hat{u}) \right] = \int_{-\infty}^{\infty} p_{u_0}(\hat{u}_i) g'(\hat{u}_i) d\hat{u}_i = \frac{1}{M} \int_{-\infty}^{\infty} p_{u_0}(\hat{u}_i) \alpha(\delta(\hat{u}_i - \theta) + \delta(\hat{u}_i - \hat{\theta})) d\hat{u}_i = 0. \tag{5.101}
\]
In the following we assume equal noise power \( \sigma_n^2 \) at each of the sensors, but uncorrelated noise signals, so that the sensor noise vector is described by \( \mathcal{N}(0, \sigma_n^2 \mathbf{I}) \), or by \( R_n = \sigma_n^2 \mathbf{I} \). This is a reasonable assumption, as very often noise is of thermal origin, therefore given by the temperature and the noise figure and as such of equal variance but mutually uncorrelated for all the channels. Furthermore, the noise power \( \sigma_n^2 \) is presumed to be known, be that from theoretical calculations of thermal noise or by estimating it, e.g., using minor component analysis in an overdetermined separation case [74]. For identical distributions of all source signals, Eq. (5.89) can be simplified to
\[
W_{t+1} = W_t + \mu \left( \mathbf{I} - g(\mathbf{u}) \mathbf{u}^T + \sigma_n^2 b W_t R_n W_t^T \right) W_t, \tag{5.102}
\]
where
\[
b = E \left[ g'(\hat{u}) \right] = \frac{2}{3-\theta^2}. \tag{5.103}
\]
For the uniform distribution, which is a good approximation for \( M \)-ary distributions when \( M \) is high, with unit variance, implying that the threshold function is properly scaled according to Eq. (5.11) we get
\[
b = E \left[ g'(\hat{u}) \right] = \frac{2}{3-\theta^2}. \tag{5.104}
\]
If the source signals have discrete distributions rather than continuous ones, the update equation, Eq. (5.102), is not accurate, as it is based on the assumption of uniform distribution. Owing to its discrete distribution, \( p_{u_0}(\cdot) \), which should be used in Eq. (5.101), is a probability mass function (pmf) rather than a pdf. However, close to the real solution we may approximate \( p_{u_0}(\cdot) \) by \( p_{u_1}(\cdot) \), which is a true pdf. Since the noise passed through to the outputs determines the probability density at the threshold level \( \theta \), \( p_{u_0}(\theta) \) depends on the separation matrix. For an \( M \)-ary signaling scheme (e.g., \( M \)-PAM) we can write for the probability density at the \( i \)th output
\[
p_{u_i}(\theta) = \frac{1}{M} \frac{1}{\sqrt{2\pi \sigma_n}} \frac{1}{\sqrt{\sum_{k=1}^{M} w_{ik}^2}}. \tag{5.105}
\]
with \( w_{ik} \) being the \((i,k)\)th element of the separation matrix \( W \), describing the path from the \( k \)th sensor to the \( i \)th output and \( \theta \) given by Eq. (5.35). For \( M \)-PAM signals, using Eqs. (5.36) and (5.105) in Eq. (5.101) and the update equation, Eq. (5.89), we get

\[
W_{t+1} = W_t + \mu \left( I - g(u)u^T + \frac{2(M+1)}{3\pi(M-1)} \sigma_n^2 \sigma_n \left( \text{diag} (W_tW_t^T) \right)^{-\frac{1}{2}} W_tW_t^T \right) W_t,
\]

(5.106)

where \( \text{diag} (W_tW_t^T) \) means here the matrix \( W_tW_t^T \) with suppressed off-diagonal terms and may also be written by the use of the Hadamard or Schur product:

\[
\text{diag} (W_tW_t^T) \equiv \bigodot (W_tW_t^T),
\]

where \( \bigodot \) denotes element-wise multiplication.

### 5.7.2 MMSE vs. zero-forcing solution

Very often in data communications we are not interested in the solution of \( W \) that directly inverts \( A \)—the so-called zero-forcing solution—due to problems associated with noise enhancement at frequencies close to zeros of the system transfer function. In terms of signal purity—the essence of low bit-error rates—we do not care where unwanted contributions to the signal come from; signals from other channels or thermal noise. This is of course only the case if channels are not jointly detected. For single-channel detection, the proper criterion to choose is the minimum mean square error (MMSE). If we have a zero-forcing solution \( W_{ZF} \), we can, by looking at the MMSE solution for unit-power source signals \[74\]

\[
W_{\text{MMSE}} = A^T (AA^T + \sigma_n^2 I)^{-1},
\]

(5.107)

reformulate the MMSE solution in terms of the zero-forcing solution. To this end, we note that the zero-forcing solution is the inverse of the system matrix but for some permutation

\[
W_{ZF} = PA^{-1}.
\]

(5.108)

Using Eq. (5.108) in Eq. (5.107) leads to

\[
W_{\text{MMSE}} = P^T W_{ZF} (W_{ZF}^{-1} P P^T W_{ZF} - \sigma_n^2 I)^{-1}.
\]

(5.109)

Of course, \( PP^T = I \), and by premultiplying the solution in Eq. (5.109) by \( P \) we do not challenge its validity, so we get

\[
W_{\text{MMSE}} = W_{ZF} (W_{ZF}^{-1} W_{ZF} - \sigma_n^2 I)^{-1}.
\]

(5.110)

For a similar method to Eq. (5.110) see also \[43\].

### 5.7.3 Computer simulations

In the following, results of computer simulations \[99\] of the blind separation using the bias-removal method suggested above are shown. Some important parameters influencing the performance were taken from \[49\], such as the number of sources and sensors \( M_s = 3 \), the mixing matrix

\[
A = \begin{bmatrix}
0.4 & 1.0 & 0.7 \\
0.6 & 0.5 & 0.5 \\
0.3 & 0.7 & 0.2
\end{bmatrix},
\]

(5.111)

and the condition \( W_0W_0^T = 0.25 \cdot I \) (implying that \( W_0 \) is a scaled orthogonal matrix) for the one hundred trials with a different initial separation matrix. In the first experiment three uniformly distributed source signals were mixed, and noise was added at the sensors with \( \sigma_n^2 = 0.01 \). The noise level was assumed to be known to the algorithm. The mixed noisy signals were then separated using the threshold nonlinearity with \( \theta = 1.5 \) and the update equation, Eq. (3.17). The step size was adjusted without noise to obtain an interchannel interference level of \(-35\text{dB}\) and then fixed to \( \mu = 0.00032 \) for the other simulations. The performance measure used in the plots is calculated as a function of the global system matrix \( P = [p_{ik}] \) from Eq. (3.4) and expresses the average interchannel...
interference. Fig. 5.17 reveals the convergence improvement of the modified algorithm compared to the standard algorithm without bias removal. Since the mixing matrix and the noise power are identical to the parameters chosen in [32] and [49], we can directly compare the convergence with those results. The curves shown in Fig. 5.17 look almost identical to the curves given in [32] and [49]. The advantage of this method here lies in the application of a much simpler nonlinearity, essentially a three-level quantizer.

Still better results were obtained for binary signals. Three binary source signals were mixed using the same mixing matrix as above. To show clearer differences between the algorithms, the noise was increased by 5 dB, resulting in $\sigma_e^2 = 0.0316$. Fig. 5.18 shows that the modified algorithm is in fact capable of completely removing any bias, albeit at a lower convergence speed. Again, step sizes were chosen equal ($\mu = 0.0018$) for all three cases. It was also observed that the modified algorithm with certain noise levels (e.g., $\sigma_e^2 = 0.01$) consistently outperformed the standard algorithm without noise. This surprising effect is due to an increased stability region (see Fig. 5.7) for lower SNRs.

The additive noise has then a positive *dithering* effect. With other nonlinearities (e.g., $g(u) = au^3$) or other distributions (e.g., uniform distribution), this effect cannot be observed. It is only the special arrangement of the high derivative of the nonlinearity at the level of spikes in the pdf that benefits from additional noise.

![Figure 5.18: Separation convergence of bias removal algorithm for binary distributions.](image)

### 5.8 Summary

A simple threshold nonlinearity is able to separate sub-Gaussian signals, e.g., M-ary PAM and QAM. The threshold function is not just a simplification of polynomial functions but a clipped version of the true score function for the uniform distribution. The convergence speed for sub-Gaussian signals is comparable to that of other known nonlinearities. The threshold function offers very simple implementation options, since the set of possible output values of this function only contains three values, ±$a$ and zero for sub-Gaussian signals, and two values, ±$a$ for super-Gaussian signals within the family of generalized Gaussian distributions, respectively. The threshold operation can therefore easily be implemented by two comparators only.

Stability analyses reveal that the threshold nonlinearity is robust in the sense that a threshold value can be found such that deviation from this value is uncritical. They also show that for threshold values larger than one, separation of sub-Gaussian signals is achieved. If the threshold is reduced to zero, super-Gaussian signals can be separated. Using the kurtosis of the output signal to train the threshold parameter, a new, computationally much simpler method than existing ones can be devised for the blind separation of mixed-kurtosis signals. On average it is equally fast as the fastest known, but more complex, methods and can therefore be implemented at a lower cost without sacrificing performance.

A fundamental result is the existence of a threshold value for any non-Gaussian distribution, making the threshold nonlinearity a valid choice for the nonlinear function used in adaptive blind signal separation.

When discrete distributions are modeled by a summation of Gaussian kernels, stability of BSS algorithms can be proven for such distributions typical in communication systems. Algorithmic extensions reduce biasing, which is introduced by additive white Gaussian noise present in the transmission channel. Under certain circumstances, additive noise might even improve convergence properties if bias removal techniques are properly applied. From an unbiased separation solution, which satisfies a zero-forcing criterion, an MMSE solution can be readily obtained by simple matrix operations. The methods for bias removal shown in this chapter can easily be extended to complex quadrature signals. [100] gives some hints as to how the update equations have to be modified.

It is relatively straightforward to modify the blind separation of instantaneously mixed source signals using the threshold nonlinearity to do single-
channel or multichannel blind deconvolution. For an efficient implementation, the methods described in [75] may be applied.

Chapter 6

Blind equalization/deconvolution

In this chapter we leave the subject of blind separation and turn to blind equalization, the problem of deconvolving two signals that are convolved with each other. Many techniques used in BSS are also applicable to blind deconvolution. After introducing algorithm classification criteria and formulating the problem in Section 6.1 and Section 6.2, respectively, we quickly review the traditional Bussgang-type algorithms such as the Sato algorithm and the constant-modulus algorithm (CMA) in Section 6.3. These methods can be accelerated using techniques introduced in Section 6.4. All these algorithms are designed for sub-Gaussian signals. Section 6.5 deals with the extension of the Sato algorithm, the CMA, and other Bussgang-type algorithms to equalize super-Gaussian signals. We show a local stability analysis of the original and the extended algorithms followed by simulation results supporting the theory. Section 6.6, finally, reviews approaches similar to the natural-gradient update equation of the previous chapters, adopted for blind equalization.

6.1 Introduction

Historically, blind equalization (BE), also referred to as self-recovering or self-training equalization, has been around longer than blind separation. Whether the reason for this is the lack of potential applications or the difficulty of the latter is hard to answer. As to the difficulty of blind deconvolution tasks, Larry Paarmann [108] wrote five years ago:
Whereas few undergraduates put much stock in the idea that twisting and mixing prescribed by convolution is a reasonable way to compute an output from an input, even fewer of them would buy the notion that this mess could be untangled, or “deconvolved”, to recreate the input.

... we would completely lose our credibility with the first-time convolver by suggesting the possibility of “blind” deconvolution.

Which of the two tasks—blind signal separation or blind deconvolution (equalization)—is considered more difficult depends on what aspect is looked at (see [50]). Blind separation is more difficult in that more than one distinct distribution—possibly of different kurtosis sign—might be present, whereas with blind equalization the source has just one—possibly known—distribution. Blind equalization is more difficult if the perfect-inversion feasibility is an issue. Whereas with separation, where perfect source recovery is usually possible provided the mixing matrix is invertible, Benveniste et al. [18] early on pointed out that for the equalization of a nonminimum-phase system, the inverse can never be described by finitely many FIR parameters. Fractionally spaced (FS) equalizers can often circumvent this problem. For FS equalizers, however, the justification for using HOS is often lost, since SOS may suffice in some situations. Such approaches are outside the scope of this thesis. Instead, we focus again on methods that make implicit use of higher-order statistics by applying some nonlinear function to the output signal of the equalizer filter.

6.1.1 Motivation

As long as communication systems such as GSM can devote up to 22% of their transmission time to pilot tones [105], there is a huge potential for blind equalizers. Moreover, propagation environments and system considerations may make the use of training sequences impractical. Besides, system designs involving heavy coding, thus working in low raw BER environments, prohibit the use of data-aided equalizers. An example of such a system is the new standard¹ for HDTV in the US [55].

¹Although the standard as such does not specify the receiver equalizer, the system specification is such that blind equalizers are clearly favored [55].

6.1.2 Classification of algorithms

Although the terms “blind deconvolution” (BD) and “blind equalization” are often used interchangeably, “blind deconvolution” is used in a more general way. The word “deconvolution” often refers to the reverse operation of the convolution of two signals without explicitly stating which of the two signals is of interest. Thus, deconvolution means either identification, where the system parameters are of interest, or equalization, where the input signal to the system is the desired information.

Very often, blind equalization algorithms are classified according to the location of their nonlinearity in the algorithm chain. Proakis and Nikias [115], for example, distinguish between three different types:

- polyspectral algorithms
- Bussgang-type algorithms
- probabilistic algorithms

In the first method the nonlinearity sits at the output of the channel, right before the equalizer filter. The nonlinearity has thus the function of estimating the channel and feeding that information to the equalizer for adapting the filter taps. For further literature see for example [15] or [64].

The second type consists of methods that have the nonlinearity at the output of the equalizer filter. Nonlinearities are often memoryless and underlie the Bussgang property, see Eq (4.1). Because such methods have often shorter convergence times than polyspectral methods, which need larger amounts of data for an equivalent estimation variance, they are more popular. Existing literature will be listed in the next section.

For the third class of methods it is more tricky to directly locate the nonlinearity. Rather, the nonlinearity is combined with the data detection process. These methods can extract most information from little data, but often at a huge computational cost. An excellent tutorial of such techniques and other blind equalization methods can be found in [127].

In a related manner, Haykin [29] suggests to distinguish between HOS methods, cyclostationary-statistics methods, and information-theoretical methods.
6.2 Problem formulation

6.2.1 Model and assumptions

Blind equalization is a task frequently used in systems that prohibit the use of training sequences, either to save capacity or because the nature of the signal does not provide an opportunity to embed known sequences into the signal, e.g., acoustic applications [56]. Fig. 6.1 shows a configuration of such a system, where the source signal $s$ is filtered by the transmission channel designated by $h(z)$. Some noise may be added at this point. The convolved signal $x$ is then deconvolved by the deconvolution filter $w(z)$ to recreate the source signal $u$. If $w(z)$ can invert $h(z)$, $u$ is close to the original source signal $s$, possibly time delayed and flipped in sign. Since the channel response $h(z)$ is unknown, $w(z)$ has to be estimated blindly. This is usually done in an adaptive way, using a stochastic-gradient method to minimize some blind cost function.

In vector notation, a blind equalization system can be described as follows. Let the vector $h = [h_0, h_1, \ldots, h_L]^T$ denote the coefficients of the channel filter of length $L + 1$. We will assume this channel to be constant, but unknown. By filling up the time series $s$ into a vector of the form $s_t = [s_t, s_{t-1}, \ldots, s_{t-L}]^T$, we can describe the input to the equalizer as

$$x_t = h^T s_t + n_t,$$

where the noise vector is designated as $n_t$. Likewise, we have $w_t = [w_{-N,t}, \ldots, w_{0,t}, \ldots, w_{N,t}]^T$ denoting the coefficients of the deconvolution filter of length $2N + 1$. Using a symmetric configuration we imply an initialization of the filter coefficients of $w_{k,0} = \delta(k)$, essentially a center-tap initialization, which allows the causal development of inverse filters for nonminimum-phase systems. Since the equalizer will be an adaptive filter, we have the additional parameter $r$, indicating that $w_t$ is the set of coefficients at time $t$. Defining the vector of channel outputs as $x_t = [x_{t+N}, \ldots, x_t, \ldots, x_{t-N}]^T$, we get the signal after the equalizer as

$$u_t = w_t^T x_t.$$ 

The blind equalization task is now to find vector $w$ such that the signal $u_t$ is essentially the same as the source signal $s_t$. By “essentially” we mean up to some ambiguities plus terms relating to the noise $n_t$. In the case of blind equalization these ambiguities are the magnitude, the phase, and the delay. The two first ambiguities are shared with the BSS problem (cf Section 3.1.1). The additional delay ambiguity can be controlled to some extent by the initialization of the equalizer coefficients. However, other local solutions of the blind cost function exist that are shifted versions of the equalizer taps, in which the equalizer might get trapped.

Alternatively, the formulation of the above task can be carried over to the $z$-domain. By assigning $s(z), n(z), x(z), u(z)$ the $z$-transforms of the source signal, the noise signal, the channel output, and the equalizer output, respectively, and

$$h(z) = \sum_{k=0}^L h_k z^{-k},$$

$$w(z) = \sum_{k=-N}^N w_{k,N} z^{-k},$$

the $z$-transforms of the channel filter and the equalizer, respectively, we can write

$$x(z) = h(z)s(z) + n(z)$$

and

$$u(z) = w(z)x(z) = w(z)h(z)s(z) + w(z)n(z)$$

for the input to and the output of the equalizer, respectively.

6.2.2 Performance measure

In the BD case, the performance measure should reflect the deconvolution capability, hence indicating to what extent the deconvolved signal is influenced by
adjacent samples of the same signal (convolutive noise). We define the overall system response as
\[ p(z) = w(z)h(z) = \sum_{k=-N}^{N+L} p_k z^{-k}, \]
where \(2N+L\) is the order of the global impulse response. Similarly to Eq. (3.4), the intersymbol interference (ISI) can be defined as
\[ J_{\text{ISI}}(p(z)) \triangleq \frac{\sum_{k=-N}^{N+L} |p_k|^2 - \max_k |p_k|^2}{\max_k |p_k|^2} - 1. \]
In order to achieve small intersymbol interference \(J_{\text{ISI}}(p(z))\), the polynomial \(p(z)\) has to get close to \(p(z) = p_d z^{-d}\), i.e., only the coefficient for one power of \(z\) is essentially different from zero.

6.3 Traditional Bussgang-type algorithms

6.3.1 The Sato algorithm

Sato was the first researcher to propose a blind equalizer for communication signals [117]. He had an application in mind, where transmission routes might be reconnected, thus incurring drastic changes of transmission conditions. He proposed the following update equation for a linear blind equalizer
\[ w_{t+1} = w_t + \mu (\gamma \text{sign}(u) - u) x, \]
which, if interpreted as the stochastic-gradient update associated with the mean square of the error signal
\[ e_5 \triangleq g(u) - u \]
for \(g(u) = \gamma \cdot \text{sign}(u)\), minimizes the cost function
\[ J(w) \triangleq E \left\{ e_5^2 \right\} = E \left\{ (\gamma \text{sign}(u) - u)^2 \right\} \]
\[ = E \left\{ |u| - \gamma \right\}^2. \]

Note that we mean again \(x \triangleq x_t\) and \(u \triangleq u_t\). The Sato parameter \(\gamma\) is defined as
\[ \gamma \triangleq \frac{E \left\{ s^2 \right\}}{E \left\{ |s| \right\}}, \]
also known as the dispersion constant. For a BPSK signal, the Sato algorithm is equivalent to a decision-directed equalizer. For \(M\)-ary signaling with \(M > 2\), minimization of the error in the MSB (most significant bit) is achieved.

6.3.2 Godard’s algorithm

Godard [60] came up with a more general expression for the cost function
\[ J(w) \triangleq E \left\{ (|u|^p - R_p)^2 \right\}, \]
a special case of which (for \(p = 1\)) is the Sato algorithm. The Godard parameter \(R_p\) is defined as
\[ R_p \triangleq \frac{E \left\{ |s|^p \right\}}{E \left\{ |s|^p \right\}}, \]
and acts as a scaling parameter (see also [100]) in the same way as the Sato parameter \(\gamma\); it also normalizes the output of the equalizer.

6.3.3 The constant-modulus algorithm (CMA)

The CMA is the most often used algorithm for blind equalization [55]. It is a special case of Godard’s algorithm and was independently of Godard discovered by Treichler and Agee [126]. The cost function is Eq. (6.13) for \(p = 2\), which leads to the following update equation
\[ w_{t+1} = w_t - \mu u(|u|^2 - R_2) x^* \]
The gradient of the CM cost surface is dependent on the source kurtosis. Loosely speaking, for many model mismatches, the CMA cost function will behave similarly to an MMSE criterion [71]. The CM criterion for 16-QAM is illustrated in Fig. 6.2. The circle is the reference curve, from which any deviation will add up to the CM measure \(J_{\text{CMA}}\). Despite the fact that 16-QAM is not a constant-modulus constellation, the CM measure \(J_{\text{CMA}}\) still maintains its minimum (> 0) at perfect equalization, and grows with increasing distortion. As can be further seen, the CM measure is phase insensitive.
Figure 6.2: Constellation and CMA cost function of some distorted 16-QAM signals. Upper left: original signal; lower left: heavily distorted; upper right: equalized with low residual distortion; lower right: equalized but with phase error.

6.3.4 The Benveniste-Goursat-Ruget (BGR) algorithm

Benveniste et al. [18] came up with a couple of fundamental results, which are valid for most blind algorithms. They show, for example, that in order to achieve blind deconvolution, the tap weights of the equalizer have to be adjusted such that the instantaneous distribution of the output converges to the input distribution.

Similarly to the generalization of the CMA by Godard’s algorithm, the Sato algorithm might be generalized, resulting in [18]

\[ w_{t+1} = w_t + \mu (\gamma \text{sign}(u) - f(u)) x, \]  

(6.16)

where

\[ \gamma = \frac{E[fx]}{E[s^2]}, \]  

(6.17)

Eq. (6.16) is what later became known as the BGR algorithm [42].

6.3.5 Benveniste-Goursat’s algorithm

The Benveniste-Goursat algorithm [17], not to be confused with the BGR algorithm introduced in the last section, is a smooth combination of the Sato and a decision-directed algorithm. The error function is given as

\[ e_{BG} \triangleq k_1 e_M + k_2 | e_M | e_S, \]  

(6.18)

where \( e_S \) is the Sato error from Eq. (6.10) and

\[ e_M \triangleq Q(u) - u, \]  

(6.19)

where \( Q(.) \) is a quantizer suitable for the detection of the modulation scheme used. Because the errors are combined, a simple equivalent cost function can in most cases not be given. For large \( e_M \) (initial phase), the second term of Eq. (6.18), hence the Sato error, is dominant. For an open-eye condition, the first term with the decision-directed error dominates, yielding a higher resolution in the resulting error signal. Deshpande [39] reports still better performance if the Sato error in Eq. (6.18) is replaced by the CMA error.

6.3.6 The Stop-and-Go Algorithm

Another combination of the robustness of a blind error function and the resolution of the decision-directed error function is the Stop-and-Go algorithm, initiated by Picchi and Prati [113]. The error function can be written as

\[ e_{SG} \triangleq \frac{1}{2} e_M + \frac{1}{2} | e_M | \text{sign}(e_S), \]  

(6.20)

which is indeed very similar to Eq. (6.18). The main difference between Eqs. (6.18) and (6.20) is that the two terms in Eq. (6.20) cancel if the signs of \( e_S \) and \( e_M \) differ. In this case, the algorithm stops adjusting, hence its name.

Proposed variants of the Stop-and-Go Algorithm include Hatzinakos’ algorithms [63], where instead of the Sato error in Eq. (6.20) he used the CMA error, and in a yet improved version, he incorporated both errors to decide if the algorithm is stopped or not. Further combined algorithms include the ones found by Weerackody and Kassam [134], and Kim et al. [82].
### 6.3.7 Other criteria

Li and Mbarek [89] defined a cost function that is best described as the closeness to a discrete-valued signal (DVS) [128] whose constellation points are given by \( a_m, m = 1, \ldots, M \)

\[
J_{\text{DVS}} = \prod_{m=1}^{M} (a - a_m)^2. \tag{6.21}
\]

The corresponding error signal is obtained by differentiation of Eq. (6.21) with respect to the filter coefficients

\[
e_{\text{DVS}} = 2 \sum_{m=1}^{M} \left( (a - a_{m1}) \prod_{m_{2} \neq m_{1}}^{M} (a - a_{m2})^2 \right). \tag{6.22}
\]

If a switch to decision-directed algorithms is desired, phase synchronization such as achieved through individual nonlinearities for I and Q are vital.

### 6.3.8 Summary of algorithms

In many blind adaptive algorithms, a cost function \( J(w) \) can be defined. In general \( J(w) \neq E[|e|^2] \) but \( e \) is chosen from

\[
\frac{\partial J(w)}{\partial w} = -ex. \tag{6.23}
\]

This way, the general update equation

\[
w_{t+1} = w_t + \mu \frac{\partial J(w_t)}{\partial w} = w_t + \mu ex \tag{6.24}
\]

is valid. Table 6.1 lists the algorithms most commonly used for adaptive blind equalization.

### 6.4 Increased convergence speed

As with nonblind stochastic-gradient algorithms, the choice of the step size is a compromise between convergence speed and excess noise. The situation gets
more problematic if we have a time-varying environment, because too small a step size (to achieve low excess noise) may lead to tracking lag. Increased convergence speed can be achieved by one of several methods:

- switching to decision-directed methods (see Fig. 6.3)
- data reusing (block iterations)
- step-size control
- combination of cost functions
- gradient modification (Newton, quasi-Newton, self-orthogonalizing, natural gradient)

![Receiver](Image)

**Figure 6.3:** Equalizer training modes: training-sequence based (1), blind (2), decision directed (3).

The Newton method needs the knowledge of the Hessian matrix, which is usually not known, and besides, the needed inversion is computationally prohibitive. The quasi-Newton method only requires an estimate of the gradient vector. For a quadratic cost function, the Newton approach results in a self-orthogonalizing approach, since the Hessian matrix is simply the autocorrelation matrix, see also [136]. The autocorrelation matrix can be calculated as

\[ R_{xx}(t + 1) = \lambda R_{xx}(t) + (1 - \lambda)x_t x_t^H, \tag{6.25} \]

and is used in \( R_{xx}(t)^{-1} x_t \), which replaces the regression vector \( x_t \) in the update equation. Near convergence we have [47]

\[ W' (z^{-1}) W(z) \approx S_{xx}^{-1}(\Omega), \tag{6.26} \]

so that by replacing the autocorrelation matrix estimate by the double convolution we get the natural-gradient approach. In Section 6.6 we will have a brief look at natural-gradient methods.

### 6.5 An extension to Bussgang-type algorithms

#### 6.5.1 Introduction

The Sato algorithm and the CMA were originally designed for the blind equalization of digital communication signals, which are usually sub-Gaussian distributed. When applied to super-Gaussian signals, failure of convergence is reported [14]. In the following we give an explanation as to why these algorithms are unsuitable for super-Gaussian signals, followed by a proposed modification to make the algorithms applicable to super-Gaussian signals.

#### 6.5.2 Local stability analysis

Many Bussgang-type algorithms such as the ones given above can be written in the form

\[ w_{t+1} = w_t + \mu(g_1(u_t) - g_2(u_t)) x_t, \tag{6.27} \]

where \( g_1(\cdot) \) and \( g_2(\cdot) \) are different nonlinearities. Without loss of generality we assume the source signal has unit power. However, the output power given by the stationary point of Eq. (6.27) need not necessarily be of unit power. For the local stability analysis we now express the deviation of the equalizer coefficients from their optimal values as

\[ w_t = w_{t, \text{opt}} + \Delta_t. \tag{6.28} \]

The output of the equalizer

\[ u_t = \sigma_d x_t - J + \sum_i \Delta_i x_{t-i}, \tag{6.29} \]
is expressed as a scaled version of the original signal sequence (with a possible delay of \( d \) samples) perturbed by convolutive noise resulting from the coefficient deviations, where \( \sigma_n \) allows for a different output energy. In the following we assume a two-sided infinitely long equalizer impulse response in order to circumvent any problems associated with finite-length truncation. The summation limits of the last section are therefore changed to \( \pm \infty \). The perturbation vector \( \mathbf{\Delta} = [\Delta_{-\infty}, \ldots, \Delta_0, \ldots, \Delta_{\infty}]^T \) now propagates similarly to Eq (6.27) as

\[
\mathbf{\Delta}_{t+1} = \mathbf{\Delta}_t + \mu \left( g_1(u_{t+d}) - g_2(u_{t+d}) \right) \\
\quad = \mathbf{\Delta}_t + \mu \left( g_1(\sigma_n u_{t-d} + \sum_k \Delta_k x_{t-k}) - g_2(\sigma_n u_{t-d} + \sum_k \Delta_k x_{t-k}) \right) x_t,
\]

where we have made use of Eq (6.29). To facilitate the analysis, we approximate the nonlinearities \( g_1() \) and \( g_2() \) by their respective first-order Taylor series around \( \sigma_n u_{t-d} \)

\[
g_{1,2} \left( \sigma_n u_{t-d} + \sum_i \Delta_i x_{t-i} \right) \approx g_{1,2}(\sigma_n u_{t-d}) + \frac{\partial g_{1,2}}{\partial u} (\sigma_n u_{t-d}) \sum_i \Delta_i x_{t-i}.
\]

Using Eq. (6.31) in Eq. (6.30) and taking the expectation on both sides yields

\[
E\{\mathbf{\Delta}_{t+1}\} = E\{\mathbf{\Delta}_t\} + \mu E \left\{ (g_1(\sigma_n u_{t-d}) - g_2(\sigma_n u_{t-d})) x_t \right\} \\
\quad + \mu E \left\{ (g_1'(\sigma_n u_{t-d}) - g_2'(\sigma_n u_{t-d})) \sum_i \Delta_i x_{t-i} x_t \right\}.
\]

At a stationary point of the update equation given by Eq. (6.27) we have

\[
E\left\{ (g_1(\sigma_n u_{t-d}) - g_2(\sigma_n u_{t-d})) x_t \right\} = 0,
\]

the condition that also determines the output power, so that we are left with the last term in Eq. (6.32), the difference of the derivatives. The convergence behavior of Eq. (6.32) can be analyzed using the ordinary differential equation method (ODE) [19]. Doing so we get an ODE of the form

\[
\frac{d\mathbf{\Delta}}{dt} = \mu A \mathbf{\Delta}.
\]

6.5. An extension to Bussgang-type algorithms

The elements of the transition matrix are obtained by looking at

\[
\frac{d\mathbf{\Delta}_j}{dt} = \mu E \left\{ g_1(\sigma_n s_{t-d}) - g_2'(\sigma_n s_{t-d}) \right\} \sum_k \Delta_k x_{t-k} x_{t-l}.
\]

Thus, the matrix \( \mathbf{A} \) may be expressed as

\[
\mathbf{A} = \begin{bmatrix}
\cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
\end{bmatrix}
\]

where the nonlinear moments are given by

\[
\zeta_0 \triangleq E \left\{ (g_1'(\sigma_n s) - g_2'(\sigma_n s)) s^2 \right\},
\]

\[
\zeta_1 \triangleq E \left\{ g_1'(\sigma_n s) - g_2'(\sigma_n s) \right\} E \left\{ |s|^2 \right\} = E \left\{ g_1'(\sigma_n s) - g_2'(\sigma_n s) \right\}.
\]

The last equality in Eq. (6.40) stems from the unit-variance assumption. Alternatively, \( \mathbf{A}_1 \) may be written as

\[
[A_1]_{jk} = \zeta_1 \left[ \sum_f h_{j-f} h_{d-k} \right]_{jk} + (\zeta_0 - \zeta_1) \left[ h_{j-f} h_{d-k} \right]_{jk},
\]

\[i, k = -\infty, \ldots, 0, \ldots, \infty.\]

The matrix \( \mathbf{A} \) is both infinite dimensional and symmetric. Symmetric matrices have real eigenvalues. A solution is said to be locally stable if all eigenvalues of
\( \mu A \) are negative. The matrix \( \mu A \) is then called negative definite. Unlike with the analysis of BSS systems, where only coupling between two coefficients occurs at a time, here the full matrix has to be analyzed. By defining the channel matrix

\[
H = \begin{bmatrix}
\vdots & \vdots & \vdots & \vdots \\
... & h_d & h_{d+1} & h_{d+2} \\
... & h_{d-1} & h_d & h_{d+1} \\
... & h_{d-2} & h_{d-1} & h_d \\
\vdots & \vdots & \vdots & \vdots
\end{bmatrix}
\]

(6.42)

and the channel vector \( \bar{h} = [h_d, h_{d-1}, h_{d-2}, \ldots] \), respectively, we can formulate \( \bar{A} \) as a sum of an autocorrelation matrix and an outer product

\[
\bar{A} = \xi_1 \bar{H}^T \bar{H} + (\xi_0 - \xi_1) \bar{h} \bar{h}^T.
\]

(6.43)

The bar in \( \bar{h} \) indicates the reverse order of its elements compared to some earlier definition of \( h \). As an infinite-dimensional matrix, \( H \) can be regarded as circulant with all its corresponding properties. The eigenvalues of \( H \) are then just the DFT coefficients of a row of \( H \) \[61\]. Hence, if the transfer function of the channel does not have spectral nulls, \( H \) is nonsingular and thus, \( H^T H \) is positive definite. The matrix \( \bar{h} \bar{h}^T \) is positive semidefinite and is of rank one. Sufficient conditions for \( \mu A \) to be negative definite are therefore

\[
\mu \xi_1 < 0,
\]

(6.44)

\[
\mu \xi_0 \leq \mu \xi_1.
\]

(6.45)

Many Bussgang-type algorithms can be written as a difference of two polynomials

\[
w_{t+1} = w_t + \mu (a \text{sign}(u_t)) |u_t|^\gamma - \text{sign}(u_t) |u_t| \cdot \varphi(x_t),
\]

(6.46)

which can be regarded as a special form of Eq. (6.27) with

\[
g_1(u) = a \text{sign}(u) |u|^\gamma,
\]

(6.47)

\[
g_2(u) = \text{sign}(u) |u| \cdot \varphi.
\]

(6.48)

Note that the scaling factor \( a \) allows an adjustment of the output power at the stationary point. The respective nonlinear moments for some known Bussgang equalizers are summarized in Table 6.2 (general output power) and Table 6.3 (normalized output power). The output power \( \sigma_w^2 \) can be obtained from the
6.5. An extension to Bussgang-type algorithms

The evaluation of the stationary point, given by the third column of Table 6.2, results in the stability conditions given in Table 6.4.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Stability conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sato</td>
<td>$\mu &gt; 0$, $2\gamma p_0(0) &lt; 1$</td>
</tr>
<tr>
<td>CMA</td>
<td>$\mu &gt; 0$, $\kappa_4 &lt; 0$</td>
</tr>
<tr>
<td>Godard</td>
<td>$\mu &gt; 0$, $\xi_1 &lt; 0$</td>
</tr>
<tr>
<td>Monomials</td>
<td>$\mu \xi_0 &lt; 0$, $\mu \xi_1 &lt; 0$</td>
</tr>
</tbody>
</table>

Table 6.4: Stability conditions for Bussgang-type equalization algorithms as a result of the nonlinear moments given by Table 6.2.

In the following we are going to show that for Bussgang-type algorithms of the form given by Eq. (6.46), sufficient stability conditions can be decoupled with respect to $\xi_0$ and $\xi_1$. To this end, we need some results from Chapter 2. We can then show the following:

**Lemma 6.1**

For a Bussgang-type equalizer in the form of a difference of two monomials according to Eq. (6.46), sufficient decoupled stability conditions on the nonlinear moments are

\[ \mu \xi_1 < 0, \quad \mu \xi_0 < 0. \]

**Proof:** For the nonlinear moments of Eq. (6.46) we get by using Eqs. (6.39) and (6.40) and still assuming unit-variance source signals

\[ \xi_0 = a_0 p_0 \sigma_u^{p-1} E \left[ |s|^{p+1} \right] - q_0 \sigma_u^{q-1} E \left[ |s|^{q+1} \right], \]

\[ \xi_1 = a_1 p_0 \sigma_u^{p-1} E \left[ |s|^{p-1} \right] - q_0 \sigma_u^{q-1} E \left[ |s|^{q-1} \right]. \]

From the evaluation of the stationary point of Eq. (6.46) we get

\[ \sigma_u = \left( \frac{E \left[ |s|^{p+1} \right]}{E \left[ |s|^{q+1} \right]} \right)^{1/p}. \]
Thus, using Eq. (6.53) in Eqs. (6.51) and (6.52) we have

\[ \xi_0 = a \frac{\frac{q+1}{p}}{\frac{q+1}{p}} E \left[ \frac{|s|^{p+1}}{|s|^{q+1}} \right] \left( p - q \right), \]  

\[ \xi_1 = a \frac{\frac{q+1}{p}}{\frac{q+1}{p}} E \left[ \frac{|s|^{p+1}}{|s|^{q+1}} \right] \left( p E \left[ |s|^{p+1} \right] - q E \left[ |s|^{q+1} \right] \right), \]  

as already listed in Table 6.2. The scaling factor \( a \) must be positive, otherwise no equilibrium can be reached. Assume for the time being that \( \mu > 0 \). In order for \( \xi_0 \) to be smaller than zero, we must choose \( p < q \). Inside the bracket term of Eq. (6.55) \( p \) gets multiplied by an equal or higher value than \( q \) due to Corollary 2.1, which implicates that \( E \left[ |s|^{m-1} \right] / E \left[ |s|^{m+1} \right] \) is a nonincreasing function in \( m \) (reciprocal). Thus, we have

\[ \mu \xi_0 \leq \mu \xi_1. \]  

A similar argument can be made for negative \( \mu \); we end up with the same inequality, Eq. (6.56).

### 6.5.3 Stability for generalized Gaussian distributions

Now, let us have a closer look at the algorithms provided in Table 6.2. The first three algorithms are known to be stable update equations for sub-Gaussian signals. For super-Gaussian signals such as Laplacian distributed signals, the Bussgang-type algorithms fail to provide local stability. In the following we prove a theorem for a general Bussgang-type equalizer in the form of the difference of two monomials. We use again an intermediate result from Chapter 2 to show that Bussgang-type algorithms generally do not work for super-Gaussian signals.

**Theorem 6.1**

No blind Bussgang-type algorithm of the general form

\[ w_{t+1} = w_t + \mu (a \text{sign}(u_t)|u_t|^p - \text{sign}(u_t)|u_t|^q) x_t \]  

(6.57)

can be stabilized by any choice of \( p \) and \( q \) and any super-Gaussian distribution.

**Proof:** Due to Lemma 6.1, the nonlinear moments \( \xi_0 \) and \( \xi_1 \) need to have the same sign. Then, stability can be controlled by proper choice of the sign of the step size \( \mu \). Without loss of generality, we assume that \( p < q \), which makes \( \xi_0 < 0 \), as can be seen from Table 6.2. But due to Lemma 2.2, we note that

\[ \frac{p \Gamma \left( \frac{q}{a} \right)}{\Gamma \left( \frac{p+2}{a} \right)} - \frac{q \Gamma \left( \frac{p}{a} \right)}{\Gamma \left( \frac{q+2}{a} \right)} < 0, \]  

making \( \xi_1 > 0 \) (see again Table 6.2), thus enforcing a different sign of \( \xi_1 \). \( \square \)

### 6.5.4 Algorithm modifications

A possible approach of applying the Sato algorithm to super-Gaussian signals using prewhitening and reversing a sign in the algorithm has been reported by Benveniste [18] (see also [15]). In other words, although inverting the sign of the step size \( \mu \) might help satisfy one of the two stability conditions, the second one might still not be satisfied. As a possible method to stabilize Bussgang-type algorithms we suggest to append individual penalizing factors of the equalizer coefficient norm to the two nonlinearities \( g_1(\cdot) \) and \( g_2(\cdot) \). To keep the algorithms fairly general we consider update equations of the form obtained by extending Eq. (6.27) to

\[ w_{t+1} = w_t + \mu (\|w_t\|^p g_1(u_t) - \|w_t\|^q g_2(u_t)) x_t \]  

(6.59)

By approximating

\[ \|w + \Delta\|^p = \left( \sum_k (w_k + \Delta_k)^2 \right)^{p/2} \approx \|w\|^p + \|\Delta\|^p \|w\|^{p-2} \sum_k w_k \Delta_k, \]  

(6.60)
the first-order Taylor series approximation of the scalar part of the update term becomes
\[
\| w \|^2 g_1 \left( \sigma_0 s_{t-d} + \sum_i \Delta_i x_i \right) - \| w \|^2 g_2 \left( \sigma_0 s_{t-d} + \sum_i \Delta_i x_i \right) \\
\approx \| w \|^2 g_1 (\sigma_0 s_{t-d}) + \| w \|^2 g_2 (\sigma_0 s_{t-d}) \sum_i \Delta_i x_i \\
+ \tilde{p} \| w \|^2 \sum_i \Delta_i w_i g_1 (\sigma_0 s_{t-d}) \\
- \| w \|^2 g_2 (\sigma_0 s_{t-d}) - \| w \|^2 g_2 (\sigma_0 s_{t-d}) \sum_i \Delta_i x_i \\
- \tilde{q} \| w \|^2 \sum_i \Delta_i w_i g_2 (\sigma_0 s_{t-d}). \tag{6.61}
\]

The additional norm factors \( \| w \|^2 \) and \( \| w \|^2 \) in Eq. (6.59) shift the stationary point from its original solution to one where the output power \( \sigma_o^2 \) does not necessarily match the original power of the source distribution. Thus, at the stationary point we get
\[
E \left\{ \left( \| w \|^2 g_1 (\sigma_0 s_{t-d}) - \| w \|^2 g_2 (\sigma_0 s_{t-d}) \right) x_i \right\} = 0, \tag{6.62}
\]
where the power of the output signal is given by the combined transfer function of the channel and the equalizer as
\[
\sigma_o^2 = \| w \|^2 \sigma_e^2 = \| w \|^2. \tag{6.63}
\]
Eqs. (6.62) and (6.63) define a system of equations for the two unknown \( \sigma_o \) and \( \| w \|^2 \). Going through a similar analysis as in the above case we end up with an ODE of the form
\[
\frac{d \Delta}{dt} = \mu A \Delta = \mu (A_1 + A_2) \Delta, \tag{6.64}
\]
where \( A_1 \) is given by Eq. (6.38), and
\[
A_2 = \begin{bmatrix}
\vdots & \tilde{\xi}_1 h_{d+1} w_{-1} & \tilde{\xi}_1 h_{d+1} w_0 & \tilde{\xi}_1 h_{d+1} w_1 & \cdots \\
\vdots & \tilde{\xi}_1 h_{d} w_{-1} & \tilde{\xi}_1 h_{d} w_0 & \tilde{\xi}_1 h_{d} w_1 & \cdots \\
\vdots & \tilde{\xi}_1 h_{d} w_{-1} & \tilde{\xi}_1 h_{d} w_0 & \tilde{\xi}_1 h_{d} w_1 & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots
\end{bmatrix}. \tag{6.65}
\]

The nonlinear moments in \( A_1 \) are now different from the ones given by Eqs. (6.39) and (6.40)
\[
\zeta_0 \triangleq \| w \|^2 E \left[ g_1 (\sigma_o s) x_i \right] - \| w \|^2 E \left[ g_2 (\sigma_o s) x_i \right], \tag{6.66}
\]
\[
\zeta_1 \triangleq \| w \|^2 E \left[ g'_1 (\sigma_o s) \right] - \| w \|^2 E \left[ g'_2 (\sigma_o s) \right]. \tag{6.67}
\]
Furthermore, we have introduced
\[
\tilde{\zeta}_1 \triangleq \tilde{p} \| w \|^2 \sum_i \Delta_i w_i g_1 (\sigma_0 s_{t-d}) - \tilde{q} \| w \|^2 \sum_i \Delta_i w_i g_2 (\sigma_0 s_{t-d}). \tag{6.68}
\]
The short-hand notation leads to
\[
[A]_{ik} = \zeta_1 \left[ \sum_j h_{j+k-i} h_j \right]_{ik} + (\zeta_0 - \zeta_1) \left[ h_{d-i} h_{d-k} \right]_{ik} + \tilde{\zeta}_1 \left[ h_{d-i} w_k \right]_{ik}, \tag{6.69}
\]
where we have defined
\[
\tilde{h} \triangleq (\zeta_0 - \zeta_1) h + \tilde{\zeta}_1 w. \tag{6.70}
\]
Although the two last terms of Eq. (6.69) are two outer products, the sum has still only one nonzero eigenvalue owing to the outer products sharing the same column vector. Hence, the outer product \( hh^T \) is either positive semidefinite or negative semidefinite with all but one eigenvalues equal to zero. Its definiteness depends on the single nonzero eigenvalue, which is
\[
\lambda = \text{tr} \left( hh^T \right) = h^T h. \tag{6.71}
\]
The first part of Eq. (6.64) requires again
\[
\mu \zeta_1 < 0. \tag{6.74}
\]
For a negative semidefinite expression for the second part of Eq. (6.64), we must ensure that
\[
\mu h^T \tilde{h} = \mu h^T (\zeta_0 - \zeta_1) h + \tilde{\zeta}_1 w \leq 0. \tag{6.75}
\]
Without loss of generality we can restrict the channel response to unit energy, thus \( \hat{h}^T \hat{h} = 1 \). We designate \( w_{\text{opt}} \) as the solution of the equalizer coefficients that perfectly inverts the channel \( \hat{h} \), i.e., \( \hat{h}^T w_{\text{opt}} = 1 \), so that the output power is \( \sigma_y^2 = ||h * w_{\text{opt}}||^2 = 1 \). Because we have added a norm factor in the update equation, we can no longer expect the output power \( \sigma_y^2 \) to be controllable purely by adjusting the gain parameter \( a \). Moreover, \( \sigma_y^2 \) is depending on the channel impulse response, or more accurately, on the norm of the equalizer impulse response that inverts this channel. The equalizer taps are adjusted to a scaled version of \( w_{\text{opt}} \).

\[
\mathbf{w} = \sigma_y \mathbf{w}_{\text{opt}}.
\]  

(6.76)

Subsequently, we also have

\[
||\mathbf{w}||_m^m = \sigma_y^m ||w_{\text{opt}}||^m. \tag{6.77}
\]

Therefore, the convolution of the channel impulse response and the equalizer impulse response yields

\[
\hat{h}^T \mathbf{w} = \sigma_y. \tag{6.78}
\]

Hence, the second condition for negative definiteness of \( M_A \) equates to

\[
\mu(\xi_0 - \xi_1 + \tilde{\xi}_1 \sigma_y) \leq 0. \tag{6.79}
\]

For a modified version of an algorithm according to Eq. (6.59), Eqs. (6.74) and (6.79) are sufficient conditions for local stability. In summary, the first stability condition, given by Eq. (6.74), stays the same for the algorithm modification, although the nonlinear moment \( \xi_1 \) changes. We can satisfy Eq. (6.74) by a proper choice of the sign of the step size \( \mu \). If this choice results in the failure of the previous second stability condition as given by Eq. (6.50), we have now an additional term \( \xi_1 \) in Eq. (6.79) to stabilize the algorithm.

### 6.5.5 Examples

In the following we look at three modified algorithms in more detail and describe sufficient conditions for local stability.

**Extended Sato algorithm**

If for some distribution we wish to equalize, the first stability condition in Eq. (6.49) is dissatisfied, i.e., \( \mu(2p_s(0)/E[|s|] - 1) \geq 0 \), we may satisfy it by choosing a negative step size \( \mu \). This results inevitably in the failure of norm convergence, since the second condition, i.e., Eq. (6.50) is now hurt. We can modify the Sato algorithm by appending the norm factor \( ||w||^\tilde{p} \) to the update equation and hence get

\[
\mathbf{w}_{t+1} = \mathbf{w}_t + \mu(||w||^\tilde{p} \gamma \text{sign}(u_t) - u_t) x_t. \tag{6.80}
\]

By determining the stationary point of Eq. (6.80) and using Eq. (6.77) we get the standard deviation of the output signal

\[
\sigma_y = ||w_{\text{opt}}||^\frac{1}{1+p} (\gamma E[|s|])^{\frac{1}{1+p}} \tag{6.81}
\]

and the norm relation

\[
||w||_m^m = ||w_{\text{opt}}||_m^m (\gamma E[|s|])_m^m. \tag{6.82}
\]

In order to test this extended algorithm for local stability we need to compute the nonlinear moments. Using the definitions in the previous sections we get

\[
\xi_0 = -1, \tag{6.83}
\]

\[
\xi_1 = ||w||^\tilde{p} 2\gamma p_s(0) - 1
\]

\[
= ||w||^\tilde{p} 2\gamma \frac{p_s(0)}{\sigma_y} - 1
\]

\[
= ||w_{\text{opt}}||^\tilde{p} \frac{2\gamma}{\sigma_y} 2\gamma \frac{p_s(0)}{\sigma_y} - 1
\]

\[
= \frac{2p_s(0)}{E[|s|]} - 1, \tag{6.84}
\]

\[
\tilde{\xi}_1 \sigma_y = \tilde{p} ||w||^\tilde{p} - 2\gamma E[|s|] \sigma_y
\]

\[
= \tilde{p} ||w_{\text{opt}}||^\tilde{p} \frac{2\gamma}{\sigma_y} 2\gamma E[|s|] ||w_{\text{opt}}||^\tilde{p} (\gamma E[|s|])^{\frac{1}{1+p}}
\]

\[
= \tilde{p} ||w_{\text{opt}}||^{2\tilde{p}}. \tag{6.85}
\]

For negative \( \mu \), the stability condition in Eq. (6.74) now means

\[
\frac{2p_s(0)}{E[|s|]} > 1, \tag{6.86}
\]

which is exactly the contrary of the stability condition of the original Sato algorithm, while the stability condition in Eq. (6.79) can be written as

\[
||w_{\text{opt}}||^2 \leq \tilde{p} E[|s|]. \tag{6.87}
\]
General Bussgang-type algorithm

In the following we would like to state the sufficient conditions for a Bussgang-type algorithm of the general extended form of Eq. (6.46). We modify the algorithm by appending a norm factor to each of the two nonlinearities. Our new update equation is thus

$$w_{t+1} = w_t + \mu(\|w_t\|^2a_{\text{sign}(u_t)}|u_t|^p - \|w_t\|^2\text{sign}(u_t)|u_t|^q)x_t.$$  (6.88)

The nonlinear moments necessary for the computation of the stability regions are

$$\zeta_0 = a\|w\|^p q_{\sigma_a}^{-1}E\left\{s^{p+1}\right\} - \|w\|^q q_{\sigma_a}^{-1}E\left\{s^{q+1}\right\},$$  (6.89)

$$\zeta_1 = a\|w\|^p q_{\sigma_a}^{-1}E\left\{s^{p+1}\right\} - \|w\|^q q_{\sigma_a}^{-1}E\left\{s^{q-1}\right\},$$  (6.90)

$$\tilde{\zeta}_1 = a\|w\|^p q_{\sigma_a}^{-1}E\left\{s^{p+1}\right\} - \|w\|^q q_{\sigma_a}^{-1}E\left\{s^{q+1}\right\}. $$  (6.91)

where

$$\sigma_a = \|w_{\text{opt}}\|^p \left(\frac{\alpha}{E\{s^{p+1}\}}\right)^{\frac{1}{p}} \left(\frac{\beta}{E\{s^{q+1}\}}\right)^{\frac{1}{q}}.$$  (6.92)

Using Eq. (6.93) in Eq. (6.77) yields

$$\|w\|^m = \left(\frac{\alpha}{E\{s^{p+1}\}}\right)^{\frac{m}{p}} \|w_{\text{opt}}\|^\frac{m}{p} \left(\frac{\beta}{E\{s^{q+1}\}}\right)^{\frac{m}{q}}.$$  (6.94)

With this, the nonlinear moments given in Eqs. (6.89) to (6.91) can be decoupled from $\|w\|$ and be written as a function of $\|w_{\text{opt}}\|$, giving

$$\zeta_0 = \|w_{\text{opt}}\|^p \left(\frac{\alpha}{E\{s^{p+1}\}}\right)^{\frac{1}{p}} \left(\frac{\beta}{E\{s^{q+1}\}}\right)^{\frac{1}{q}} (p - q),$$  (6.95)

$$\zeta_1 = \|w_{\text{opt}}\|^p \left(\frac{\alpha}{E\{s^{p+1}\}}\right)^{\frac{1}{p}} \left(\frac{\beta}{E\{s^{q+1}\}}\right)^{\frac{1}{q}} (p - q) E\{s^{q-1}\},$$  (6.96)

$$\tilde{\zeta}_1 = \|w_{\text{opt}}\|^p \left(\frac{\alpha}{E\{s^{p+1}\}}\right)^{\frac{1}{p}} \left(\frac{\beta}{E\{s^{q+1}\}}\right)^{\frac{1}{q}} (\tilde{p} - \tilde{q}).$$  (6.97)

For a negative step size $\mu$ the first stability condition, given by Eq. (6.74), means that

$$\frac{E\{s^{p+1}\}}{E\{s^{q+1}\}} > -\frac{q}{p},$$  (6.98)

whereas the second stability condition, given by Eq. (6.79), now turns into

$$\|w_{\text{opt}}\|^2 \leq \frac{\tilde{p} - \tilde{q}}{q \left(1 - \frac{E\{s^{p+1}\}}{E\{s^{q+1}\}\}} - p \left(1 - \frac{E\{s^{q+1}\}}{E\{s^{q+1}\}\} \right).$$  (6.99)

At this place it should be pointed out again that these stability conditions are sufficient, but not necessary, since we stated that some components of the matrix $A$ must be positive (semi)definite, whereas in reality only the sum of this components, namely $A$ itself needs to be positive definite.

Extended Constant-Modulus Algorithm (CMA)

In the third example we extend the CMA with a norm in the first nonlinearity,

$$w_{t+1} = w_t + \mu(\|w\|^p R_{2u_t} - u_t^2)x_t.$$  (6.100)

This is just a special case of the more general one we discussed in the previous section. We may therefore derive the stability conditions directly from Eq. (6.99) using $p = 1$, $q = 3$, and $\tilde{q} = 0$. Condition (6.74) for the CMA requires $\left(1 - \frac{3}{E\{|s|^4\}}\right) > 0$ or $E\{|s|^4\} > 3$, which is simply a requirement on the kurtosis to be positive. Discarding common factors in the nonlinear moments, Condition (6.99) for the CMA results in

$$\|w_{\text{opt}}\|^2 \leq \frac{\tilde{p} - \tilde{q}}{3} E\{|s|^4\} - 1.$$  (6.101)

6.5.6 Computer simulations

The channel model used in this section for the computer simulation of two extended Bussgang-type algorithms and a Laplacian distributed source signal is a 7-tap FIR filter with the impulse response $h(z) = 0.4 + \frac{z^{-1}}{} - 0.7z^{-2} + 0.6z^{-3} + 0.3z^{-4} - 0.4z^{-5} + 0.1z^{-6}$. Its properties are depicted in Fig. 6.4. Moderate noise has been added resulting in SNR=30 dB.
A computer simulation was carried out to verify the stable behavior of the extended Sato algorithm and the extended CMA for a super-Gaussian signal (Laplacian distribution). The step size was chosen as $\mu = 0.0006$ for all algorithms. For the extended Sato algorithm, we used $p = 2$, $q = 0$, and for the extended CMA $p = 4$, $q = 0$, respectively. Figs. 6.5 and 6.6 show the results of this simulation. Both the Sato algorithm and the CMA fail to converge, as would be expected for this type of distribution, whereas the extended algorithms converge as predicted by the theory. The norms of the extended algorithms converge towards values different from that of the optimal inverting filter. The stable norm values depend on the choice of the norm exponents and the channel.

Figure 6.4: Channel model for simulation of extended Sato algorithm; a) zero-pole plot, b) impulse response, c) frequency response (magnitude), d) frequency response (phase).

Figure 6.5: Convergence of the original and the extended Bussgang-type algorithms.

Figure 6.6: Equalizer norm development of the original and the extended Bussgang-type algorithms.
6.6 Natural-gradient updates

The use of “fast” gradients was briefly mentioned in Section 6.4. Since such modified-gradient algorithms exist for the blind BSS, see Chapter 3, it is only natural to try to adopt these approaches to the BE problem. Douglas and Haykin [50] transform the matrix $A$ into a circular matrix representing the coefficients of the channel impulse response $h(z)$. Since the inverse of a circular matrix is circular itself, we get, by taking a row computation of the BSS algorithm, a natural-gradient update equation for the BE problem. The interpretation of the RHS term $g(u)^T W_t$ in the natural-gradient update equation transformed to the blind equalization case is ambiguous. The right-multiplication by the matrix $W_t$ corresponds to a convolution by the filter $w_t$. This convolution can be carried out in two different ways [45]. Either the conjugate output of the filter, $u^*$, is correlated by the equalizer impulse response, or the output of the nonlinearity, $g(u)$, is convolved by the equalizer impulse response, yielding two possible update equations for the $k$th filter tap

$$w_{k,t+1} = w_{k,t} + \mu \left( w_{k,t} - g(u_{t-L}) \sum_{l=0}^{L} w_{t-l,t-l} u_{t-k-l}^* \right)$$ \hspace{1cm} (6.102)

and

$$w_{k,t+1} = w_{k,t} + \mu \left( w_{k,t} - u_{t-L}^* \sum_{l=0}^{L} w_{t-l,t-l} g(u_{t-l}) \right),$$ \hspace{1cm} (6.103)

respectively. It must be noted here that with these transformations from circular matrices to finite-length filters, the equivariant performance property gets lost [50].

Although natural-gradient based algorithms—as all Bussgang-type algorithms shown in this thesis—produce a biased solution for the coefficient estimation in noisy environments similar to what was shown in Section 5.7.1, this is usually not a problem in equalization, since often the filter switches over to a decision-directed tracking mode.

6.6.1 The threshold nonlinearity

As is shown in [131], the stability conditions for the BSS and the BE problem, expressed as a function of the source distribution and the nonlinearity chosen, are essentially the same. Thus, all the stability regions developed in Chapter 4 and Chapter 5 are also applicable to the BE problem.

As an example, we use the threshold nonlinearity to blindly equalize the channel given in Fig. 6.4, excited by a uniformly distributed signal. A comparison of the convergence time with that of a cubic nonlinearity reveals a comparable performance, as can be seen in Fig. 6.7.

![Figure 6.7: Convergence curves for natural-gradient blind equalization algorithm.](image)

6.7 Summary

Bussgang-type algorithms are a popular choice for the blind equalization of signals. Digital communication signals, usually sub-Gaussian distributed, allow the utilization of a variety of combinations of purely blind methods and decision-directed approaches, either in a hard-changing manner or in a soft changeover. Furthermore, there is a selection of methods that increase the convergence speed of Bussgang-type algorithms, some of which use decision-directed methods, hence might be called semiblind.
Sufficient local stability conditions for the standard Bussgang-type algorithms such as the Sato algorithm and the CMA reveal the reasons for the failure of these algorithms for the family of super-Gaussian distributions. Modifications in the form of additional norm factors of the filter coefficients in the update equations stabilize these algorithms for distributions where the original algorithms fail.

A slightly different route can be taken by relating the BSS algorithms discussed in previous chapters to adaptive equalizer algorithms. The same theory of natural-gradient algorithms in the sense of local stability regions as discussed earlier applies here.

Chapter 7

Blind carrier phase synchronization

Blind adaptive phase synchronization is a related topic to blind adaptive signal separation. As seen in Chapter 4, if the real part and the imaginary part of the signals are treated separately in blind signal separation, phase equalization up to an ambiguity of \( \pi/2 \) can be achieved, a fact that might be exploited for phase synchronization. Starting out with a strong justification for phase synchronization, we derive the symbol error probability for vestigial sideband modulation (VSB) in the presence of phase errors and compare it with the symbol error rates for 8-PAM and 64-QAM in Section 7.2. Monte-Carlo simulations support the theoretical expressions. Phase misadjustments lead to error floors, which are computed, and which further reveal the susceptibility of VSB signals to phase errors. Phase synchronizers, as introduced in Section 7.3 can either work in a data-aided manner or blindly. We focus on the latter and show the relationship between phase synchronization and blind signal separation. Finally, a method suitable for the phase synchronization of VSB signals as used by the ATSC DTV transmission system is presented and evaluated. Its performance is measured against that of the decision-directed approaches that have been suggested in the literature. Depending on the SNR of the VSB signal, either blind or decision-directed approaches exhibit superior performance. A semi-blind method, which combines the advantages of the decision-directed and the blind mode in their respective SNR regions, is introduced and set in perspective with the other algorithms. Finally, simulation results are given in Section 7.4.
7.1 Introduction

7.1.1 Vestigial sideband modulation

Pulse-amplitude modulation (PAM) is a digital modulation scheme that is not very bandwidth efficient, since it has a double-sided, redundant spectrum. Making better use of the spectrum can be achieved in three different ways: The alphabet size might be increased, information might be added to the imaginary part of the baseband representation leading to quadrature-amplitude modulation (QAM), or one sideband might be eliminated. Vestigial sideband (VSB) modulation is one such format of the latter group, whereby a PAM signal is filtered to essentially half the bandwidth. In contrast to SSB signals, for which the suppressed sideband vanishes completely, a VSB signal has its undesired sideband eliminated through filtering. Additionally, a carrier is present, facilitating easy synchronization through a phase-locked loop (PLL). Thus, any VSB signal is generated by simply taking the Hilbert transform of the corresponding PAM signal, plus adding a carrier. 8-VSB is the modulation scheme adopted by the Grand Alliance for the high-definition television (HDTV) broadcasting standard in the US [124]. A scatter diagram of the complex baseband representation of an 8-VSB signal is given in Fig. 7.1. In the following we assume that timing synchronization is not an issue and has been solved prior to phase synchronization, an assumption often justified in practice.

Figure 7.1: Scatter diagram of 8-VSB signal. The imaginary part is roughly Gaussian distributed.

7.2 Phase-error induced symbol error rates

7.2.1 SER of PAM with phase error

The symbol error rate (SER) of an M-PAM signal can be easily obtained by considering separately the two outer and the \( M - 2 \) inner constellation points. For the former points, a symbol error can only occur if the noise component shifts the signal inwards. The chance for this to happen is given by the probability

\[
P = Q\left(\frac{\sqrt{2E_s}}{N_0 A}\right),
\]

where \( E_s \), \( N_0 \), and \( A \) are the symbol energy, the noise density, and the amplitude of the smallest constellation point, respectively, and \( Q(x) \) is the tail function\(^1\) given by

\[
Q(x) \equiv \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\left(-\frac{\xi^2}{2}\right) d\xi.
\]

To produce a unit-power signal, \( A \) must be chosen according to

\[
A = \sqrt{\frac{3}{M^2 - 1}},
\]

where \( M \) is the number of constellation points. For the inner constellation points, a decision error can happen towards either side. All possible errors accounted for and averaged over an equal symbol probability, we obtain for the symbol error probability the well-known result

\[
P_s = \frac{2}{M} \left[ M - 1 \right] \frac{E_s}{N_0} Q\left(\frac{\sqrt{3}}{N_0 A \sqrt{M^2 - 1}}\right).
\]

If a phase error \( \phi \) is present in the PAM signal, the errors made for different constellation points are no longer equally likely. Using geometrical considera-

\(^1\)Note that although it is called the tail function, \( Q(x) \) is defined for negative arguments \( x \), as well.
tions, the SER can be written as
\[ P_S = \frac{2}{M} \left( \sum_{m=0}^{M/2-1} Q \left( \frac{2E_S}{N_0} A \left( 1 - (2m+1)(1 - \cos \phi) \right) \right) + \sum_{m=M/2}^{M/2-1} Q \left( \frac{2E_S}{N_0} A \left( 1 + (2m+1)(1 - \cos \phi) \right) \right) \right). \] (7.5)

### 7.2.2 SER of QAM with phase error

In contrast to the M-PAM signal mentioned in the previous section, the baseband representation of both an \( M^2 \)-QAM signal and an \( M \)-VSB signal have nonzero imaginary parts. In the case of \( M^2 \)-QAM, an error might occur if the in-phase or the quadrature component cross the decision region of the symbol sent and any of its neighbor symbols. The SER can thus be constructed as
\[ P_S = 1 - \left( 1 - \frac{2}{M^2} \sum_{n=-M/2+1}^{M/2-1} \left( \sum_{m=0}^{M/2-1} Q \left( \frac{2E_S}{N_0} \left( A - (a_m - \sqrt{a_m^2 + b_n^2} \cos(\arctan \frac{b_n}{a_m} + \phi)) \right) \right) + \sum_{m=M/2}^{M/2-1} Q \left( \frac{2E_S}{N_0} \left( A + (a_m - \sqrt{a_m^2 + b_n^2} \cos(\arctan \frac{b_n}{a_m} + \phi)) \right) \right) \right)^2 \right). \] (7.6)

with \( a_m = A(2m+1) \) and \( b_n = A(2n+1) \) representing the discrete amplitude levels of the real and the imaginary parts of the symbols, respectively. Compared to the previous section, \( A \) is now reduced by a factor of \( \sqrt{2} \) to result in the same signal power, owing to the QAM’s two-dimensional constellation. Thus,
\[ A = \sqrt{\frac{3}{2(M^2 - 1)}}. \] (7.7)

### 7.2.3 SER of VSB with phase error

In addition to the real part (denoted by I for in-phase component) shared by the PAM signal, the VSB signal also has an imaginary part (denoted by Q for quadrature component), which is generated by the Hilbert transform of the corresponding PAM signal. For zero phase error, the symbol error probability of a VSB signal is a simple 3-dB shift of its PAM counterpart due to dividing up the original power between I and Q. For a phase error \( \phi > 0 \), however, points further away from the real axis are more exposed to decision errors due to their larger amplitude. This makes the phase sensitivity closer to that of \( 64 \)-QAM than that of \( 8 \)-PAM. To compute the SER, we must find the distribution of the imaginary part of the signal. Although the usual implementations of a Hilbert transformer comprise a finite number of coefficients, for a sufficiently high number and \( M = 8 \) constellation points, the distribution of the imaginary part can be modeled accurately enough by a normal distribution [119]. Spectrum considerations of a Hilbert signal reveal that the variance of the real and imaginary parts must be equal. With Eq. (7.7), which ensures a variance of 0.5 on the real part, the imaginary part of the signal can be modeled by a normal distribution with a variance of 0.5, too. Let us denote the Gaussian-distributed imaginary part of the VSB signal by \( b \). The expectation of the SER can now be expressed as
\[ P_S = \frac{2}{M} \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi}} e^{-b^2} \left( \sum_{m=0}^{M/2-1} \left( \sum_{n=0}^{M/2-1} Q \left( \frac{2E_S}{N_0} \left( A - (a_m - \sqrt{a_m^2 + b_n^2} \cos(\arctan \frac{b_n}{a_m} + \phi)) \right) \right) + \sum_{n=M/2}^{M/2-1} Q \left( \frac{2E_S}{N_0} \left( A + (a_m - \sqrt{a_m^2 + b_n^2} \cos(\arctan \frac{b_n}{a_m} + \phi)) \right) \right) \right)^2 \right) \right) db, \] (7.8)

with \( a_m = A(2m+1) \) representing the discrete amplitude levels of the real part of the symbols. Note that due to a symmetry in the distribution of the imaginary part of the signal, negative values of \( a_m \) do not need to be considered in Eq. (7.8), a comment equally valid for Eq. (7.6). Evaluations of Eqs. (7.5), (7.6), and (7.8) are shown in Figs. 7.2, 7.3, and 7.4, respectively, along with the results of Monte-Carlo (MC) simulations to validate the theoretical expressions. As can be readily seen, the 8-VSB signal is much more sensitive to phase errors compared to the 8-PAM signal. Its sensitivity is comparable to that of 64-QAM, although less severe, since the imaginary amplitude is concentrated more around zero, owing to its Gaussian distribution. The relationships of the SERs and the phase errors for all three modulation formats are further illustrated in Fig. 7.5.
Chapter 7. Blind carrier phase synchronization

7.2 Phase-error induced symbol error rates

Figure 7.2: Symbol error rate of 8-PAM with phase error.

Figure 7.3: Symbol error rate of 64-QAM with phase error.

Figure 7.4: Symbol error rate of 8-VSB modulation with phase error.

Figure 7.5: Symbol error rates of 8-PAM, 64-QAM, and 8-VSB modulation as a function of phase error for $E_S/N_0 = 20$ dB.
7.2.4 Symbol error floor

Even in the absence of noise, symbols might still be detected in error if the phase offset is large enough, resulting in the formation of an error floor. For the 8-PAM signal, this happens if the rotation of a constellation point due to phase error reaches the adjacent decision region. This condition can be formulated as

\[(2m + 1)(1 - \cos \varphi) = 1.\]  

(7.9)

For 8-PAM, the biggest chance is for \(m = 3\) (the outermost symbol) resulting in

\[\varphi_{\text{floor, 8-PAM}} = \arccos \left( \frac{6}{7} \right) = 0.54 \text{ rad},\]  

(7.10)

for which phase offset a symbol error floor (irreducible error rate) becomes apparent.

For 64-QAM, an error floor forms much earlier, due to the larger amplitude produced by the additional imaginary part of the signal. The error floor is determined by the corner points of the constellation, since these points show the largest amplitude. Working out the crossing point of a corner point with the decision region of its neighboring point, we get the following condition for \(\varphi\)

\[7A - 7\sqrt{2}A \cos \left( \frac{\pi}{4} + \varphi \right) > A,\]  

(7.11)

which yields the phase error at which the error floor appears

\[\varphi_{\text{floor, 64-QAM}} = \arccos \left( \frac{6}{7\sqrt{2}} \right) - \frac{\pi}{4} = 0.13 \text{ rad}.\]  

(7.12)

Since the behavior of 8-VSB is in many ways closer to that of 64-QAM than to that of 8-PAM, the error floor will be similar to the value given in Eq. (7.12). Thus, it becomes clear why phase correction for VSB signals is so important.

7.3 Residual phase-error correction

Different methods for phase-error correction of digitally modulated signals have been suggested in the literature. However, most techniques are unsuitable for VSB signals due to various reasons. A simple higher-order (HO) nonlinearity as used by Georghiades [54] to get rid of symbol contents fails, due to the indecisiveness of the imaginary part resulting in a huge variance. Although in the ideal case, all the data information can be retrieved from the real part of a VSB signal, a tentative-decision-based method does not work either if only the real part is observed, since this method cannot decide in which direction the constellation plane has been rotated. A simple cross-correlation of the real and imaginary parts of the signal is zero independent of the angle, as all second-order statistics are phase-blind.

7.3.1 Data-aided phase-error correction

Data-aided adaptive methods work well if we have “reliable” estimates of the reference signals. Lee et al. [88] use tentative decisions to update the phase error of an adaptive algorithm. Their method is particularly useful for the case of HDTV where an IF PLL removes most of the carrier uncertainty. The second phase-tracking loop, which removes any residual phase errors left from the first PLL, then operates using estimated data. One of the algorithm’s property is the stop-and-go nature, which may be compared to [113].

7.3.2 Blind phase synchronization

If for any reasons we do not wish to rely on tentative decisions as used in the method previously mentioned, we have to employ blind methods. Even for cases where decision-directed and blind approaches deliver comparable performance, blind methods have the following advantages:

- often smaller complexity
- more noise resistant (working at very high symbol error rates)
- modulation scheme need not be known
- gain fading does not affect result

In the following we show how an adaptive phase synchronizer can be derived as a special case of a blind signal separation algorithm, either of two real sources or a single, complex-valued source.
Real 2 x 2 mixing case

The in-phase and quadrature components of a complex signal may be interpreted as two real signals $\mathbf{x} = [x_1, x_2]^T$ of a linear, instantaneously mixed system of source signals $\mathbf{s} = [s_1, s_2]^T$:

$$\mathbf{x} = \mathbf{A}\mathbf{s},$$

with a restriction of the mixing matrix to

$$\mathbf{A} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix}.$$  \hfill (7.14)

$\mathbf{W}$ denotes the separation matrix that recovers the source signals, so that the output signals $\mathbf{u} = [u_1, u_2]^T$ are given by

$$\mathbf{u} = \mathbf{W}\mathbf{x}.$$  \hfill (7.15)

For sub-Gaussian signals, a blindly recovered separation matrix can be obtained by the adaptive algorithm (cf. Section 4.5)

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \mu \left( \mathbf{I} - \mathbf{u}^3 \mathbf{u}^T \right) \mathbf{W}_t,$$  \hfill (7.16)

where the nonlinearity $(\cdot)^3$ means a vector consisting of the cubes of its elements. The element-wise interpretation of Eq. (7.16) is given by

$$\begin{bmatrix} w_{11,t+1} \\ w_{12,t+1} \\ w_{21,t+1} \\ w_{22,t+1} \end{bmatrix} = \begin{bmatrix} w_{11,t} \\ w_{12,t} \\ w_{21,t} \\ w_{22,t} \end{bmatrix} + \mu \begin{bmatrix} 1 - u_1^3 \\ -u_1^2u_2 \\ u_1u_2^2 \\ 1 - u_2^3 \end{bmatrix} \begin{bmatrix} w_{11,t} \\ w_{12,t} \\ w_{21,t} \\ w_{22,t} \end{bmatrix}.$$  \hfill (7.17)

We can impose a similar constraint on $\mathbf{W}$ as on $\mathbf{A}$ by using

$$w_{11} = w_{22},$$  \hfill (7.18)

$$w_{21} = -w_{12}.$$  \hfill (7.19)

Hence, the updates of $w_{11}$, $w_{22}$ and $w_{21}$, $-w_{12}$, respectively, must be the same. We can therefore build an average update rule from Eq. (7.17)

$$w_{11,t+1} = w_{22,t+1}$$

$$= w_{11,t} + \frac{\mu}{2} \left( (1 - u_1^4) w_{11,t} - u_1^2u_2 w_{21,t} + u_1u_2^3 w_{22,t} + (1 - u_2^4) w_{11,t} \right)$$

$$= w_{11,t} + \mu \left( (1 - u_1^4/2 - u_2^4/2) w_{11,t} - (u_1^2u_2/2 - u_1u_2^3/2) w_{21,t} \right).$$  \hfill (7.20)

Similarly,

$$w_{21,t+1} = -w_{12,t+1}$$

$$= w_{21,t} + \mu \left( (1 - u_1^4/2 - u_2^4/2) w_{21,t} + (u_1^2u_2/2 - u_1u_2^3/2) w_{11,t} \right).$$  \hfill (7.21)

Eqs. (7.18) and (7.19) make the matrix $\mathbf{W}$ orthogonal but not orthonormal. The scale of $\mathbf{W}$ is given by the evaluation of the diagonal terms of Eq. (7.16)

$$\int_{-\infty}^{\infty} u^4 p(u) du = 1.$$  \hfill (7.22)

Eqs. (7.20) and (7.21) are very similar to the EASI algorithm for blind signal separation [26], but involve both signals in the update of each scaling term. A similar approach in terms of keeping the matrix $\mathbf{W}$ orthogonal, or rather orthonormal in that approach, has been presented by Sun and Douglas [122].

Complex 1 x 1 mixing case

As mentioned in Chapter 4, in a complex-valued blind signal separation algorithm, the nonlinearity can be either applied to the complex signal as a whole, or individually to real and imaginary parts [100]. If we consider a complex-valued 1 x 1 mixing case with an individually applied nonlinearity $(\cdot)^3$, we can write the update equation as

$$(w_k + j w_l)_{t+1} = (w_k + j w_l)_t + \mu \left( (1 - (u_k + j u_l^3) (u_k + j u_l)^3) \right)(w_k + j w_l)_t.$$  \hfill (7.23)

where $w_k$ and $w_l$ are the respective real and imaginary parts of the separation coefficient. Strictly speaking, we are not separating individual signals, the 1 x 1 case is more like a complex-valued AGC (automatic gain control). But by bringing the phase into the game, we may consider the situation as that of separating the real and the imaginary parts of one signal. After writing the real and imaginary parts of the separation coefficient separately, Eq. (7.23) leads to

$$w_{k,t+1} = w_{k,t} + \mu \left( (1 - u_k^4 - u_l^4) w_{k,t} - (u_k^3u_l - u_ku_l^3) w_{l,t} \right),$$  \hfill (7.24)

$$w_{l,t+1} = w_{l,t} + \mu \left( (1 - u_k^4 - u_l^4) w_{l,t} - (u_k^3u_l - u_ku_l^3) w_{k,t} \right).$$  \hfill (7.25)

Apart from some scaling, which only influences the output power after the separation process, the two update systems Eqs. (7.20)–(7.21) and Eqs. (7.24)–(7.25) are equivalent if in the 2 x 2 case two weights are assigned as stated in Eqs. (7.18) and (7.19).
Phase adaptation

In the following we assume small phase offsets \( \varphi \), common in the residual phase noise after a PLL, such as that used in [88]. By setting \( w_R = \cos \varphi \) and \( w_I = \sin \varphi \), and the substitutions \( f = (1 - u_R^4 - u_I^4) \) and \( g = (u_R^2 u_I - u_R u_I^3) \) we get by dividing Eq. (7.24) through Eq. (7.25)
\[
\tan \varphi_{t+1} = \frac{\sin \varphi_t (1 + \mu f) + \mu g \cos \varphi_t}{\cos \varphi_t (1 + \mu f) - \mu g \sin \varphi_t} 
\approx \frac{\sin \varphi_t (1 + \mu f) + \mu g \cos \varphi_t}{\cos \varphi_t (1 + \mu f)} 
= \tan \varphi + \frac{\mu g}{1 + \mu f}. 
\]
(7.26)

For small \( \mu \) Eq. (7.26) can be written as
\[
\tan \varphi_{t+1} = \tan \varphi_t + \mu (u_R^3 u_I - u_R u_I^3), 
\]
(7.27)
so we recognize that for stationary \( \varphi \), the update term must be zero in its expectation. We can now formulate the update equation of a phase tracker
\[
\hat{\varphi}_{t+1} = \hat{\varphi}_t + \mu (u_R^3 u_I - u_R u_I^3). 
\]
(7.28)

For small step sizes \( \mu \), the approximation of \( \tan \varphi \) by \( \varphi \) is well justified.

Direct phase estimation

The expectation of the expression \( (u_R^3 u_I - u_R u_I^3) \) is what interests us in the following. Assuming that \( u_R \) and \( u_I \) are the rotated version of the uncorrelated source signals \( s_R \) and \( s_I \), hence
\[
u_R = s_R \cos \varphi - s_I \sin \varphi, 
\]
(7.29)
\[
u_I = s_R \sin \varphi + s_I \cos \varphi, 
\]
(7.30)
we find after some trigonometric simplifications
\[
E{(u_R^3 u_I - u_R u_I^3)} = E{s_R^4 + s_I^4 - 6s_R^2 s_I^2} \cos \varphi \sin \varphi, 
\]
(7.31)
By assigning
\[
\gamma_s = E{s_R^4 + s_I^4 - 6s_R^2 s_I^2}, 
\]
(7.32)
which is related to the kurtosis of the in-phase and quadrature components of the source signals, we can write Eq. (7.31) as
\[
\hat{\varphi} = \frac{1}{4} \arcsin \left( \frac{4E{(u_R^2 u_I - u_R u_I^3)}}{\gamma_s} \right). 
\]
(7.33)
Chen et al. [28] have come up with a similar estimate for the phase of a QAM signal, depending on the region of the estimated phase. Cartwright [27] has taken that approach one step further and extended it beyond signals whose in-phase and quadrature component are independent making it applicable to nonsquare, cross QAM constellations (e.g., 32-QAM). His estimation equation is
\[
\tan 4\varphi = \frac{\gamma_a - \gamma_b}{\gamma}, 
\]
(7.34)
with
\[
\gamma_a \triangleq E{(u_R^4 u_I^2 - 3E{(u_R^2 u_I^2}) E{(u_R u_I)}}, 
\gamma_b \triangleq E{(u_R^2 u_I^3)} - 3E{(u_I)} E{(u_R u_I)}, 
\gamma \triangleq E{(u_I^4)} + E{(u_R^4)} - 6E{(u_R^2 u_I^2)}, 
\]
(7.35)
(7.36)
(7.37)
By replacing \( \gamma \) with \( \gamma = \gamma_s \cos 4\varphi \), which is the case under the assumption
\[
E{s_s^3 s_I} = E{s_R^3 s_I^3}, 
\]
(7.38)
we get Eq. (7.33), with the additional disadvantage that now some statistics of the source signal have to be known.

Cartwright also mentions a numerical observation of \( \gamma_s = 0 \) for M-PSK signals. This is the case for \( M > 4 \), and it can easily be verified that this is due to pairs of symbols that are \( 45^\circ \) shifts of each other, e.g., \( s_1 = 1 \) and \( s_2 = \sqrt{2}/2(1 + i) \) cancel each other in terms of contribution to \( \gamma_s \).

7.3.3 Blind phase-error correction for 8-VSB system

Existing decision-directed approach

As mentioned above, Lee et al. [88] suggest a phase-tracking loop for VSB HDTV receivers making use of tentative decisions. The phase error is estimated
using the update equation

\[ \hat{\phi}_{t+1} = \hat{\phi}_t - \mu A b_t \Delta \hat{\phi}_t, \]  

(7.39)

with

\[ b_t = \begin{cases} 
1, & \text{if } |x_t| > T, \\
0, & \text{otherwise},
\end{cases} \]  

(7.40)

\[ \Delta \hat{\phi}_t = \begin{cases} 
\Delta \hat{\phi}_{\text{max}}, & \text{if } \frac{x_t - \delta_R}{x_t} > \Delta \hat{\phi}_{\text{max}}, \\
-\Delta \hat{\phi}_{\text{max}}, & \text{if } \frac{x_t - \delta_R}{x_t} < -\Delta \hat{\phi}_{\text{max}}, \\
x_t - \delta_R, & \text{otherwise}.
\end{cases} \]  

(7.41)

The threshold \( T \) is chosen as the smallest symbol amplitude, and \( \Delta \hat{\phi}_{\text{max}} = 1 \).

The phase function \( \Delta \hat{\phi}_t \) in Eq. (7.39) is a complexity-reduced version of

\[ \Delta \hat{\phi}_t = \text{sign}(x_t) \Re \left( \arccos \left( \frac{\delta_R}{x_t} \right) - \arccos \left( \frac{x_t}{\sqrt{x_t^2 + 1}} \right) \right). \]  

(7.42)

Blind approach

If the phase function \( \Delta \hat{\phi}_t \) in Eq. (7.39) is replaced by the blind term suggested in Section 7.3.2, we get an alternative update equation

\[ \hat{\phi}_{t+1} = \hat{\phi}_t + \mu_b \frac{u_R^3 u_t - u_R u_t^3}{\gamma_s}. \]  

(7.43)

This nonlinearity works for both sub- and super-Gaussian signals due to the statistical moment \( \gamma_s \) in the denominator of Eq. (7.43). This can also be seen from the following observations. Von Hoff et al. [131] point out, that if a nonlinearity turns out to be unstable for separation due to the wrong kurtosis sign, we obtain a stable version by taking the transpose, \( u_t g(u_R) \) instead of \( g(u_t) u_R \) (see also Section 4.8). This would change the sign of the numerator of Eq. (7.43) only, so that it can be concluded that Eq. (7.43) indeed computes the right phase update regardless of the distribution. A more general form of Eq. (7.43) is possible by replacing the cubic nonlinearity by a general nonlinearity \( g(\cdot) \)

\[ \hat{\phi}_{t+1} = \hat{\phi}_t + \mu_b (g(u_R) u_t - u_R g(u_t)). \]  

(7.44)

Note that the statistical constant \( \gamma_s \) has been omitted in Eq. (7.44). It can either be absorbed (including the proper sign) by the step size \( \mu_b \) or by the gain factor \( \alpha \) of the nonlinearity. In Section 5.4 we have seen that the threshold nonlinearity can be used to perform de-rotation of complex constellations. The threshold nonlinearity is given by Eq. (5.10) as

\[ g(u) = \begin{cases} 
0, & |u| < \vartheta, \\
\alpha \text{sign}(u), & |u| \geq \vartheta.
\end{cases} \]  

(7.45)

The threshold value \( \vartheta \) is chosen for maximum sensitivity of Eq. (7.44). This can be expressed by

\[ \vartheta_{\text{opt}} = \arg \max_{\vartheta} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_{\text{R}}(s_R) p_{\text{I}}(s_I) (g(u_R) u_I - u_R g(u_I)) ds_I ds_R, \]  

(7.46)

with

\[ \begin{pmatrix} u_R \\ u_I \end{pmatrix} = \begin{pmatrix} \cos(\varphi - \hat{\phi}) & \sin(\varphi - \hat{\phi}) \\ -\sin(\varphi - \hat{\phi}) & \cos(\varphi - \hat{\phi}) \end{pmatrix} \begin{pmatrix} s_R \\ s_I \end{pmatrix}. \]  

(7.47)

Since the distributions of the real part and the imaginary part of a VSB signal differ, it makes sense to individually choose or tune the nonlinearity—or in this case the threshold value \( \vartheta \). Besides, the distribution of the real part of the VSB signal is a discrete one, whereas the imaginary part is roughly Gaussian distributed. We can thus rewrite Eq. (7.46) by

\[ \vartheta_{\text{opt}} = \arg \max_{\vartheta} \frac{1}{M} \sum_{m=1}^{M} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{s_I^2}{2 \gamma_s}} (g(u_R) u_I - u_R g(u_I)) ds_I ds_R, \]  

(7.48)

where the \( s_R \) are taken from the set

\[ \frac{3}{2(M^2 - 1)} \left\{ -M + 1, -M + 3, \ldots, -3, -1, 1, 3, \ldots, M - 3, M - 1 \right\}. \]

Numerical evaluation shows that the threshold value for the real part that maximizes Eq. (7.48) is \( \vartheta_{\text{R}} = 1.53/\sqrt{2} = 1.08 \). Furthermore, any \( \vartheta_{\text{R}} \) leads to the same value of Eq. (7.48). This is not surprising, since the Gaussian distribution is known to be inadmissible to blind methods using implicit higher-order statistics. On the contrary, using the term with the nonlinearity on the imaginary part, \( u_I g(u_R) \), just introduces excess noise on the phase estimation. It is therefore better to simply omit this term. Including \( \alpha \) and \( \gamma_s \), which is negative
for sub-Gaussian signals, into the positive step size \( \mu_b \), we get as the final expression for the blind phase-synchronization update equation for VSB signals
\[
\hat{\phi}_{t+1} = \hat{\phi}_t - \mu_b g(u_b) n_t,
\]
with the simple threshold nonlinearity
\[
g(u_b) = \begin{cases} 
-1, & u_b \leq -1.08, \\
0, & |u_b| < 1.08, \\
1, & u_b \geq 1.08.
\end{cases}
\]

In order for this adaptive phase tracking algorithm to work optimally, we have assumed that the constellation has the proper scale, i.e., some AGC stage precedes the synchronization stage. If this is not the case, a self-normalizing update equation in the form of Eqs. (7.20) and (7.21) has to be deployed. Since the decision-directed algorithm our algorithm is compared with presumes the existence of such an independent AGC stage, it seems fair to leave the scaling issue outside of the phase-tracking algorithm.

**Semiblind approach**

Compared with the decision-directed algorithm, we expect the blind approach to be better for low SNR and worse for high SNR. A combined approach, where there is a smooth changeover from one algorithm to the other would merge the advantageous SNR regions of the two algorithms. Such algorithms were common very early on in blind equalization, e.g., the Stop-and-Go Algorithm by Picchi and Prati [113] is an update equation consisting of both a blind and a decision-directed term. Following this method, we can find a semiblind algorithm for the phase-synchronization problem of the form
\[
\hat{\phi}_{t+1} = \hat{\phi}_t - \mu_{dd-b}(k_{dd}\Phi_{dd} + k_b|\Phi_b|\Phi_b),
\]
where
\[
\Phi_{dd} = b_{dd}\Delta\hat{\phi}_t,
\]
\[
\Phi_b = g(u_b)n_t.
\]

\( k_{dd} \) and \( k_b \) are constants, and \( b_{dd} \) and \( \Delta\hat{\phi}_t \) can be inferred from Eqs. (7.39) and (7.41), respectively. The mechanism of Eq. (7.51) is simple. A large decision-directed motivated phase-error term \( \Phi_{dd} \) means that the decisions are unreliable, thus emphasizing—through multiplication by its magnitude—the blind phase-error term \( \Phi_b \).

### 7.4 Simulation results

For the following simulation we have adopted most parameters from [88]. The symbol rate of the 8-VSB signal was 10.762 MHz, the bandwidth of the phase noise 60 kHz, and the standard deviation of the phase noise was \( \sigma_p = \sqrt{1.64^2} = 0.02235 \) rad. We have found that under those circumstances (the phase-error variance suggested and an SER of around 0.2) no phase tracking—in other words leaving the phase as it comes from the PLL—performed best most of the time. We thus changed some parameters to cases where doing phase tracking—be that blind or not—is more worthwhile. Compared to the original set-up of [88], we multiplied the standard deviation of the filtered phase noise by a factor of ten (now at \( 10\sqrt{1.64^2} = 0.2235 \) rad) at the same time as reducing its bandwidth by a factor of ten to 6 kHz, thereby changing the characteristic of the residual noise after the PLL. A bandwidth of 6 kHz was stated as typical by Sgrignoli et al. [119]. All other parameters were tuned for maximum performance, in detail:

\[
\mu_{dd} = 0.12, \\
\mu_b = 0.0034, \\
k_{dd} = 1, \\
k_b = 4.
\]

In the first simulation, we switched off thermal noise, since this would swamp any plot of the phase estimation. Fig. 7.6 shows the phase noise after the PLL, the estimated phase by the blind algorithm, and the residual phase noise after the blind phase synchronizer. The effect of the blind phase synchronization on the constellation can be seen by comparing Figs. 7.7 and 7.8. For the comparison of the blind and the decision-directed algorithms, we chose the SNR (or correspondingly the \( E_b/N_0 \)) of our simulation to result in a symbol error rate of about 10 to 20%, an SER range for which the succeeding trellis decoder works efficiently and the system has been specified for. The estimated phase variance is depicted in Fig. 7.9. The dependency of the variance on the SNR is strongest for the decision-directed algorithm. The blind algorithm according to Eq. (7.49) and the combined algorithm according to Eq. (7.51) yield much lower estimation variances up to an \( E_b/N_0 \) of 15 dB, the latter even lower for higher \( E_b/N_0 \). The respective symbol error rates (see Fig. 7.10) are a logical consequence of the phase-estimation performance and show that blind phase synchronization has a clear advantage over decision-directed algorithms for SERs above 10%—a result which is known from blind equalizers. The combined decision-directed/blind approach is always better than either the blind or
the decision-directed approach and comes close to the perfect phase tracking by less than 1 dB. If complexity is no issue, this semiblind algorithm is a very efficient choice.

**Figure 7.6:** Phase, phase estimate, and residual phase for the blind phase-synchronization algorithm.

**Figure 7.7:** Scatter diagram of 8-VSB signal with phase error.

**Figure 7.8:** Scatter diagram of 8-VSB signal after blind phase synchronization.

**Figure 7.9:** Variance of phase estimation vs. $E_{S}/N_0$ for different adaptive phase-correction methods.
7.5 Summary

Blind methods of phase synchronization for complex baseband signals, in particular 8-VSB as used in HDTV, are effective means to mitigate phase-offset effects. Such algorithms are strongly motivated by blind signal separation algorithms, and connections between the two areas can easily be seen. There is a clear advantage of the blind synchronization algorithms over the decision-directed counterparts in the SNR range of HDTV operation. By exploiting the superior performance of decision-directed methods in the high-SNR range, however, semiblind algorithms have proved worthwhile.

Chapter 8

Concluding remarks

8.1 Conclusions

Nonlinear functions are at the center of many adaptive blind algorithms for signal separation and equalization. Their role in Bussgang-type separation and deconvolution algorithms is to produce implicit higher-order or lower-order statistics that reveal dependencies among the signal components. The choice of this nonlinear function has a direct impact on the stability, complexity, and performance of an algorithm and depends on the source distribution. Stability conditions for a given nonlinearity can be stated. The additional stability condition for complex-valued nonlinearities and separation algorithms is automatically satisfied for sub-Gaussian distributions, so that for these distributions the well-known stability conditions of real signals are sufficient.

A key parameter characterizing source distributions is the kurtosis. Upper and lower bounds on the kurtosis of composite signals can be built and help qualify signals for their suitability to blind algorithms using the kurtosis as an independence measure.

The threshold function is a simple, yet powerful, nonlinearity for blind signal separation. The local stability of the natural-gradient blind adaptive algorithm using the threshold nonlinearity for the separation of any non-Gaussian distributed signals can be proved. By changing the threshold parameter of the threshold nonlinearity, which looks like a three-level quantizer, it can be
tuned to suit different distributions. A real-time adaptive algorithm adjusts the threshold values for unknown or changing distributions. The complexity of the proposed threshold nonlinearity is smaller than that of conventional nonlinearities such as \((x)^3\) or \(\tanh(x)\). Still, the performance of the adaptive threshold nonlinearity algorithms equals that of an algorithm changing between \((x)^3\) and \(\tanh(x)\). Besides, distributions neither separable by \((x)^3\) nor by \(\tanh(x)\) are still separable by the threshold nonlinearity. Contrary to previous belief, a universal fixed nonlinearity for the blind separation of all non-Gaussian distributed signals does not exist. The threshold nonlinearity with a variable threshold parameter comes closest to such a universal fixed nonlinearity, as it separates all non-Gaussian distributions, but it contains a single parameter.

Bussgang algorithms such as the Sato algorithm and the CMA are known to equalize sub-Gaussian signals. However, the algorithm can be extended to the space of super-Gaussian signals by an additional norm factor in the update equation of the equalizer coefficients.

Blind phase-synchronization algorithms devised by the help of complex-valued blind signal separation approaches outperform decision-directed approaches for the HDTV set-up in the US, where the operating point is in the low-SNR range.

### 8.2 Outlook

A sharp increase in research activities over the last decade indicates the viability of the ICA concept to solve blind separation problems. The field has matured considerably, and applications arise in a multitude of areas. We have but outlined a few concepts and aspects of adaptive blind signal processing. Further effort is needed to settle more detailed questions. Some aspects that it might pay to have a closer look at are:

- **Convergence speed:**
  Although the convergence speed of blind adaptive algorithms may suffice for acoustic applications, it is felt that for wireless applications with fast-changing environments increasing convergence speed of existing algorithms is vital. In particular, if both problems of separation and equalization—the so-called multichannel blind deconvolution problem—have to be solved simultaneously, the convergence properties suffer considerably.

- **Iterative methods:**
  For situations as the one given above, iterative methods might be a valid approach, since they use the existing information more economically. In its simplest form, blocks of data are repeatedly offered for training to an adaptive algorithm. Little is known about performance bounds of such techniques.

- **Complexity reduction:**
  In order to make algorithms usable in real-time environments, complexity analysis and reduction is mandatory. Together with techniques such as data reusing, faster-converging algorithms will then result for a given complexity.
Appendix A

Functions and formulae

A.1 The Gamma function

A function frequently used in connection with distributions and their moments is the Gamma function (see Fig. A.1), which is given by
\[ \Gamma(x) \triangleq \int_0^\infty \xi^{x-1} \exp(-\xi) \, d\xi. \]  
(A.1)

The Gamma function \( \Gamma() \) is a convex function for positive arguments and shows a recursive property similar to the factorial function
\[ \Gamma(x + 1) = x \Gamma(x). \]  
(A.2)

For positive integer values of \( x \) we have thus
\[ \Gamma(x) = (x - 1)!. \]  
(A.3)

Further properties of the Gamma function are [21]
\[ \Gamma(x) \Gamma(1-x) = \frac{\pi}{\sin(\pi x)}, \]  
(A.4)
\[ \Gamma(x) \Gamma(-x) = -\frac{\pi}{x \sin(\pi x)}, \]  
(A.5)
\[ \Gamma\left(\frac{1}{2} + x\right) \Gamma\left(\frac{1}{2} - x\right) = \frac{\pi}{\cos(\pi x)}. \]  
(A.6)

More information on the Gamma function can be found in [9].
Appendix A. Functions and Formulae

A.2 Stirling’s formula

Stirling’s formula [21] is an approximation for factorials of large numbers, and may therefore also be used as an approximation for the Gamma function with a large argument. It reads as

\[ x! \approx \sqrt{2\pi x} x^x e^{-x}. \]  

(A.7)

A.3 Matrix-Inversion Lemma

If \( A \) and \( C \) are nonsingular \( M \times M \) and \( N \times N \) matrices, the following equality holds [78]

\[ (A + BC D)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}. \]  

(A.8)

This is the so-called matrix-inversion lemma and can be easily verified by multiplying both sides of Eq. (A.8) by \( (A + BC D) \).

A.4 Differentiation with respect to a matrix of complex elements

Similarly, a Matrix-Inversion Lemma for block-partitioned matrices exists [80]

\[ \begin{bmatrix} A & B \\ C^T & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + E \Delta^{-1} F & -E \Delta^{-1} \\ -\Delta^{-1} F & \Delta^{-1} \end{bmatrix}, \]  

where

\[ \Delta = D - C^T A^{-1} B, \]  

(A.9)

\[ E = A^{-1} B, \]  

(A.10)

\[ F = C^T A^{-1}. \]  

(A.11)

\[ \begin{bmatrix} A & B \\ C^T & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + E \Delta^{-1} F & -E \Delta^{-1} \\ -\Delta^{-1} F & \Delta^{-1} \end{bmatrix}. \]  

(A.12)

The matrix of derivatives of a complex-valued matrix of size \( M \times N \) can be defined as follows [20]

\[ \frac{\partial}{\partial X} \equiv \frac{1}{2} \begin{bmatrix} \frac{\partial}{\partial \text{Re} x_{11}} - j \frac{\partial}{\partial \text{Im} x_{11}} & \cdots & \frac{\partial}{\partial \text{Re} x_{1N}} - j \frac{\partial}{\partial \text{Im} x_{1N}} \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial \text{Re} x_{MN}} - j \frac{\partial}{\partial \text{Im} x_{MN}} & \cdots & \frac{\partial}{\partial \text{Re} x_{MN}} - j \frac{\partial}{\partial \text{Im} x_{MN}} \end{bmatrix}. \]  

(A.13)

and

\[ \frac{\partial}{\partial X^*} \equiv \frac{1}{2} \begin{bmatrix} \frac{\partial}{\partial \text{Re} x_{11}} + j \frac{\partial}{\partial \text{Im} x_{11}} & \cdots & \frac{\partial}{\partial \text{Re} x_{1N}} + j \frac{\partial}{\partial \text{Im} x_{1N}} \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial \text{Re} x_{MN}} + j \frac{\partial}{\partial \text{Im} x_{MN}} & \cdots & \frac{\partial}{\partial \text{Re} x_{MN}} + j \frac{\partial}{\partial \text{Im} x_{MN}} \end{bmatrix}. \]  

(A.14)

After Haykin [65], the complex gradient matrix is now defined as

\[ \nabla_X \equiv 2 \frac{\partial}{\partial X^*}. \]  

(A.15)
A.5 Derivation of the relationship between the fourth-order cumulant and higher-order moments

The fourth-order cumulant has a central role in blind signal processing, not least due to its normalized variant, the kurtosis. We shall therefore derive its relationship to more accessible higher-order moments. From the definition of the $n$th-order cumulant, Eq. (2.6), we can evolve step by step

$$C_x^{(4)} \triangleq (-j)^3 \frac{d^4 K(t)}{dt^4} \bigg|_{t=0}$$

$$= \frac{d^4 \ln \phi(t)}{dt^4} \bigg|_{t=0} = \frac{d}{dt} \frac{d}{dt} \frac{d}{dt} \frac{d}{dt} \ln \phi(t) \bigg|_{t=0} = \frac{d}{dt} \frac{d}{dt} \frac{1}{\phi(t)} \frac{d}{dt} \frac{d}{dt} \phi(t) \bigg|_{t=0}$$

$$= \frac{d}{dt} \left( \frac{-1}{\phi^2(t)} \left( \frac{d\phi(t)}{dt} \right)^2 + \frac{1}{\phi(t)} \frac{d^2\phi(t)}{dt^2} \right) \bigg|_{t=0}$$

$$= \frac{d}{dt} \left( \frac{2}{\phi^3(t)} \left( \frac{d\phi(t)}{dt} \right)^2 - \frac{3}{\phi^2(t)} \frac{d\phi(t)}{dt} \frac{d^2\phi(t)}{dt^2} + \frac{1}{\phi(t)} \frac{d^3\phi(t)}{dt^3} \right) \bigg|_{t=0}$$

$$= \frac{-6}{\phi^4(t)} \left( \frac{d\phi(t)}{dt} \right)^4 + \frac{12}{\phi^3(t)} \left( \frac{d\phi(t)}{dt} \right)^2 \frac{d^2\phi(t)}{dt^2} - \frac{3}{\phi^2(t)} \left( \frac{d^2\phi(t)}{dt^2} \right)^2$$

$$- \frac{4}{\phi(t)} \frac{d\phi(t)}{dt} \frac{d^3\phi(t)}{dt^3} + \frac{1}{\phi(t)} \frac{d^4\phi(t)}{dt^4} \bigg|_{t=0}.$$  

(A.16)

With

$$\phi^m(t) \bigg|_{t=0} = 1 \quad \forall m \in \mathbb{N},$$  

(A.17)

and the definition of the moments, Eq. (2.4), we get

$$C_x^{(4)} = M_x^{(4)} - 4M_x^{(3)}M_x^{(2)} - 3 \left( M_x^{(2)} \right)^2 + 12M_x^{(2)} \left( M_x^{(1)} \right)^2 - 6 \left( M_x^{(1)} \right)^4.$$  

(A.18)

For the cases we are interested in, $p_x(.)$ is zero-mean and symmetric, hence $M_x^{(1)} = M_x^{(3)} = 0$, and thus

$$C_x^{(4)} = M_x^{(4)} - 3 \left( M_x^{(2)} \right)^2,$$  

(A.19)

which confirms Eq. (2.10).
Appendix C

Formal proofs

C.1 Proofs of the lemmas on the kurtosis of a sum of two independent signals

Proof of Lemma 2.4: We have to prove that

\[ |\kappa_4(x_1 + x_2)| < \max(|\kappa_4(x_1)|, |\kappa_4(x_2)|). \]  

(C.1)

Proof: We denote \( x_1, x_2 \) two mutually independent, zero-mean random variables with respective variance \( \sigma_1^2 \) and \( \sigma_2^2 \), and respective kurtosis \( \kappa_4(x_1) \) and \( \kappa_4(x_2) \). Without loss of generality we assume \( |\kappa_4(x_1)| \geq |\kappa_4(x_2)| \). We can then express the kurtosis of the sum \( x_1 + x_2 \) as follows (see also Eq. (2.77))

\[
\kappa_4(x_1 + x_2) = \frac{E \left( (x_1 + x_2)^4 \right)}{(\sigma_1^2 + \sigma_2^2)^2} - 3 \left( \frac{\sigma_1^2 + \sigma_2^2}{\sigma_1^2 + \sigma_2^2} \right)^2
\]

(C.2)

Note that by multiplying both sides of Eq. (C.2) by the denominator of its right-hand side, it can be readily seen that the 4th-order cumulants, as in fact all

Most SERs can be found in [114].
cumulants, just add up. To show that \(|\kappa_4(x_1 + x_2)| < |\kappa_4(x_1)|\), we need to show that
\[ -\kappa_4(x_1) < \kappa_4(x_1 + x_2) < \kappa_4(x_1), \quad \text{for } \kappa_4(x_1) > 0 \]  
(C.3)
and
\[ \kappa_4(x_1) < \kappa_4(x_1 + x_2) < -\kappa_4(x_1), \quad \text{for } \kappa_4(x_1) < 0. \]  
(C.4)

We distinguish two cases.

Case 1: \(\kappa_4(x_1) > 0\). From Eq. (C.2) we write
\[ \kappa_4(x_1 + x_2) \leq -\frac{\kappa_4(x_1)\sigma_1^4 + \kappa_4(x_2)\sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} = \kappa_4(x_1) - \frac{\sigma_1^4 + \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} < \kappa_4(x_1), \]  
(C.5)
where the first inequality in Eq. (C.5) results from \(\kappa_4(x_1) \geq \kappa_4(x_2)\), and the second inequality is valid for positive \(\kappa_4(x_1)\). From Eq. (C.2) it is clear, that \(\kappa_4(x_1 + x_2)\) is positive if \(\kappa_4(x_1)\) and \(\kappa_4(x_2)\) are both positive. Eq. (C.3) is then obvious. For negative \(\kappa_4(x_2)\) we can write
\[ \kappa_4(x_1 + x_2) > -\kappa_4(x_1) \]  
(C.6)
where the first inequality in Eq. (C.6) is again due to \(\kappa_4(x_1) \geq \kappa_4(x_2)\), and the second inequality stands because \(\kappa_4(x_2)\) is negative. By assumption \(\kappa_4(x_1) \geq -\kappa_4(x_2)\) or, equivalently, \(-\kappa_4(x_1) \leq \kappa_4(x_2)\), and with Eq. (C.6)
\[ \kappa_4(x_1 + x_2) > -\kappa_4(x_1). \]  
(C.7)

Case 2: \(\kappa_4(x_1) < 0\). From Eq. (C.2) we write
\[ \kappa_4(x_1 + x_2) \geq -\frac{\kappa_4(x_1)\sigma_1^4 + \kappa_4(x_2)\sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} = \kappa_4(x_1) - \frac{\sigma_1^4 + \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} > \kappa_4(x_1), \]  
(C.8)
where the first inequality in Eq. (C.8) results from \(\kappa_4(x_1) \leq \kappa_4(x_2)\) (absolute value of \(\kappa_4(x_1)\) is higher or equal), and the second inequality is valid for negative \(\kappa_4(x_1)\). \(\kappa_4(x_1 + x_2)\) is negative for the case where both \(\kappa_4(x_1)\) and \(\kappa_4(x_2)\) are negative. Eq. (C.4) is then satisfied. For positive \(\kappa_4(x_2)\) we write
\[ \kappa_4(x_1 + x_2) < \frac{\kappa_4(x_2)\sigma_1^4 + \kappa_4(x_2)\sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} = \kappa_4(x_2) - \frac{\sigma_1^4 + \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} < \kappa_4(x_2) \leq -\kappa_4(x_4), \]  
(C.9)

Proof of Lemma 2.5: We now prove that
\[ |\kappa_4(x_1 + x_2)| \geq \frac{\min(|\kappa_4(x_1)|, |\kappa_4(x_2)|)}{2}. \]  
(C.10)

Proof: Again we assume \(|\kappa_4(x_1)| \geq |\kappa_4(x_2)|\). Furthermore, sign(\(\kappa_4(x_4)\)) = sign(\(\kappa_4(x_2)\)). By the line of argument of the above proof we find that
\[ |\kappa_4(x_1 + x_2)| \geq |\kappa_4(x_2)| - \frac{\sigma_1^4 + \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2}. \]  
(C.11)
It can easily be verified that
\[ \frac{\sigma_1^4 + \sigma_2^4}{(\sigma_1^2 + \sigma_2^2)^2} \geq \frac{1}{2}, \]  
(C.12)
which, together with Eq. (C.11), concludes the proof. \(\square\)

C.2 A lemma on the expectation of a product of two functions

**Lemma C.1**
Assume that \(f_1(x)\) and \(f_2(x)\) are two even symmetric functions of a random variable that exhibit monotonic increasing behavior for positive \(x\). Then the expectations of these functions satisfy the inequality
\[ E\{f_1(x)f_2(x)\} \geq E\{f_1(x)\}E\{f_2(x)\}. \]  
(C.13)
Proof: We first note that if \( f_1(x) \) and \( f_2(x) \) are uncorrelated, we have equality of Eq. (C.13) by the separation property of uncorrelated functions. Furthermore, if \( f_1(x) \equiv f_2(x) \), the inequality of Eq. (C.13) can be shown by the Cauchy-Schwarz inequality. For the more general case, we make use of the interchangeability of \( f_1(x) \) and \( f_2(x) \), and assume that we can find \( x_1 \) and \( x_2 \) such that

\[
\begin{align*}
    f_1(x_1) &= E[f_1(x)] = m_1, \\
f_2(x_2) &= E[f_2(x)] = m_2,
\end{align*}
\]

with \( x_1 \leq x_2 \). Because of monotonicity and the definitions of \( x_1 \) and \( x_2 \) we have

\[
\begin{align*}
    f_1(x) - m_1 &< 0, & x < x_1, \\
    &= 0, & x = x_1, \\
    &> 0, & x > x_1,
\end{align*}
\]

and

\[
\begin{align*}
    f_2(x) - m_2 &< 0, & x < x_2, \\
    &= 0, & x = x_2, \\
    &> 0, & x > x_2.
\end{align*}
\]

It is relatively easy to show that

\[
\int_{x_2}^{\infty} p_x(x) f_1(x)(f_2(x) - m_2) \, dx \geq \int_{x_2}^{\infty} p_x(x) m_1 f_2(x) - m_2) \, dx,
\]

because in this range of the integral we have \( f_2(x) - m_2 \geq 0 \) and \( f_1(x) \geq m_1 \), always. Slightly more involved is the complementary range of the integral, which we partition in the following way

\[
\int_{0}^{x_2} p_x(x) f_1(x)(f_2(x) - m_2) \, dx = \int_{0}^{x_1} p_x(x) m_1 f_2(x) - m_2) \, dx + \int_{x_1}^{x_2} p_x(x) f_1(x) - m_1)(f_2(x) - m_2) \, dx.
\]

The second term on the RHS of Eq. (C.19) can be further split into

\[
\int_{0}^{x_1} p_x(x) (f_1(x) - m_1)(f_2(x) - m_2) \, dx
\]

\[
= \int_{0}^{x_1} p_x(x) (f_1(x) - m_1)(f_2(x) - m_2) \, dx
\]

\[
+ \int_{x_1}^{x_2} p_x(x) (f_1(x) - m_1)(f_2(x) - m_2) \, dx.
\]

We first note that clearly

\[
\int_{0}^{x_1} p_x(x)(f_1(x) - m_1) \, dx \leq 0.
\]

From the definition of the function mean we see that

\[
\int_{0}^{x_1} p_x(x)(f_1(x) - m_1) \, dx + \int_{x_1}^{\infty} p_x(x)(f_1(x) - m_1) \, dx = 0.
\]

Hence,

\[
0 \leq \int_{0}^{x_2} p_x(x)(f_1(x) - m_1) \, dx \leq -\int_{0}^{x_1} p_x(x)(f_1(x) - m_1) \, dx.
\]

By multiplying both sides of Eq. (C.23) with a number smaller than zero, which is furthermore smaller in magnitude in the interval \([x_1, x_2]\) than in \([0, x_1]\), and flipping signs we have

\[
0 \leq -\int_{x_1}^{x_2} p_x(x)(f_1(x) - m_1)(f_2(x) - m_2) \, dx
\]

\[
\leq \int_{0}^{x_1} p_x(x)(f_1(x) - m_1)(f_2(x) - m_2) \, dx.
\]

The RHS of Eq. (C.20) is thus positive so that Eq. (C.19) results in

\[
\int_{0}^{x_2} p_x(x) f_1(x)(f_2(x) - m_2) \, dx \geq \int_{0}^{x_2} p_x(x) m_1(f_2(x) - m_2) \, dx.
\]

We can now express the LHS of Eq. (C.13) as

\[
E[f_1(x)f_2(x)] = 2 \int_{0}^{x_2} p_x(x) f_1(x) f_2(x) \, dx + 2 \int_{x_2}^{\infty} p_x(x) f_1(x) f_2(x) \, dx
\]

\[
= 2 \int_{0}^{x_2} p_x(x) f_1(x) f_2(x) \, dx + 2 \int_{x_2}^{\infty} p_x(x) f_1(x) f_2(x) \, dx
\]

\[
+ 2 \int_{0}^{x_1} p_x(x) f_1(x)(f_2(x) - m_2) \, dx
\]

\[
+ 2 \int_{x_1}^{\infty} p_x(x) f_1(x)(f_2(x) - m_2) \, dx.
\]
Appendix C. Formal proofs

\begin{align*}
&= E\{f_1(x)\} E\{f_2(x)\} \\
&\quad + 2 \int_0^{\infty} p_\alpha(x) f_1(x) (f_2(x) - m_2) \, dx \\
&\quad + 2 \int_{-\infty}^{x_2} p_\alpha(x) f_1(x) (f_2(x) - m_2) \, dx \\
&\geq E\{f_1(x)\} E\{f_2(x)\} \\
&\quad + 2 \int_0^{x_2} p_\alpha(x) m_1 (f_2(x) - m_2) \, dx \\
&\quad + 2 \int_{-\infty}^{x_2} p_\alpha(x) m_1 (f_2(x) - m_2) \, dx \\
&= E\{f_1(x)\} E\{f_2(x)\} + m_1 E\{f_2(x) - m_2\} \\
&= E\{f_1(x)\} E\{f_2(x)\}, \quad (C.26)
\end{align*}

with equality only for \(|x|\) essentially constant, i.e., the binary distribution. This concludes the proof. \(\square\)

Alternatively, Eq. (C.13) may be shown as a special case of Čebyšev’s inequality [111].

C.3 Proof of Eq. (2.34)

Eq. (2.34) can be proved trivially for super-Gaussian signals by Eq. (2.44). A more general proof of Eq. (2.34) for all symmetric distributions with unit variance can be constructed as follows.

Proof: The proof is based on induction. As the basis of the induction we assume that we have shown

\begin{align*}
E\{|\bar{x}|^{m-2}\} \geq E\{|\bar{x}|^{m-4}\} \geq 1, \quad (C.27)
\end{align*}

which is true for \(m = 4\) by the assumption of unit variance. Reformulation of Eq. (C.27) leads to

\begin{align*}
\int_0^1 (x^{m-2} - x^{m-4}) p_\alpha(x) \, dx + \int_{-1}^0 (x^{m-2} - x^{m-4}) p_\alpha(x) \, dx \geq 0. \quad (C.28)
\end{align*}

Due to negative values in the integrand, the first integral of Eq. (C.28) will be negative. By multiplying the first integrand with values smaller than or equal to one, and the second integrand with values greater than or equal to one, we can write

\begin{align*}
\int_0^1 x^2(x^{m-2} - x^{m-4}) p_\alpha(x) \, dx + \int_{-1}^0 x^2(x^{m-2} - x^{m-4}) p_\alpha(x) \, dx \geq 0 \quad (C.29)
\end{align*}

or

\begin{align*}
\int_{-1}^0 x^m p_\alpha(x) \, dx \geq \int_{-1}^0 x^{m-2} p_\alpha(x) \, dx. \quad (C.30)
\end{align*}

which for symmetric distributions is equivalent to Eq. (2.34). Equality applies only if \(\bar{x}\) is restricted to \(\pm 1\), hence the binary case. \(\square\)

An alternative proof can be led using the Cauchy-Schwarz inequality.

C.4 Proof of Eqs. (2.44) and (2.45)

In the following we will make use of the \([\cdot]\) and \([\cdot]\) operators to denote the integer not greater than and the integer not smaller than, respectively. We assume even integers \(m \geq 4\), so that \([\frac{m}{2}] = [\frac{m}{2}] = \frac{m}{2}\).

Proof: We can write Eq. (2.33) as

\begin{align*}
\frac{E\{|\bar{x}|^m\} \Gamma \left( \frac{m+1}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} \right)}{E\{|\bar{x}|^{m-2}\} \Gamma \left( \frac{m-1}{\alpha} \right) \Gamma \left( \frac{3}{\alpha} \right)} = \frac{\Gamma \left( \frac{3}{\alpha} + \frac{m-2}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} \right)}{\Gamma \left( \frac{1}{\alpha} + \frac{m-2}{\alpha} \right) \Gamma \left( \frac{3}{\alpha} \right)}. \quad (C.31)
\end{align*}

Consider \(0 < \alpha < 2\) first. Using the recursive property \(\Gamma(\alpha + 1) = \alpha \Gamma(\alpha)\) (see Appendix A.1) we can lower bound Eq. (C.31) by

\begin{align*}
\Gamma \left( \frac{3}{\alpha} + \frac{m-2}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} \right) &\geq \Gamma \left( \frac{3}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} + 1 \right) \cdot \Gamma \left( \frac{3}{\alpha} \right) \\
&= \frac{\Gamma \left( \frac{3}{\alpha} \right)}{\Gamma \left( \frac{3}{\alpha} + m-2 \right) \cdot \Gamma \left( \frac{1}{\alpha} + \frac{m-2}{\alpha} \right)} \\
&= \prod_{k=0}^{\left\lfloor \frac{m-2}{\alpha} \right\rfloor -1} \frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k}. \quad (C.32)
\end{align*}

Using

\begin{align*}
\frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k} &\geq \frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k} = \frac{3 + 2k}{1 + 2k}, \quad \alpha < 2, k \geq 0, \quad (C.33)
\end{align*}

\(\square\)
with equality if and only if \( k = 0 \), and
\[
\left\lceil \frac{m-2}{\alpha} \right\rceil \geq \left\lceil \frac{m}{2} - 1 \right\rceil = \frac{m}{2} - 1, \quad \alpha < 2,
\]
we can write
\[
\frac{E[|\hat{x}|^m]}{E[|\hat{x}|^{m-2}]} \geq \prod_{k=0}^{\left\lfloor \frac{m-2}{\alpha} \right\rfloor} \frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k} \geq \prod_{k=0}^{\left\lfloor \frac{m}{2} - 1 \right\rfloor} \frac{3 + 2k}{1 + 2k} = 3 + 2\left( \frac{m}{2} - 2 \right) = m - 1,
\]
where the second but last equality is by realizing that the denominator factors are cancelled by previous numerator factors, so that the last numerator factor remains, and the last equality is by the assumption of even \( m \), which completes the proof of Eq. (2.44).

Now we consider \( \alpha > 2 \). Similarly to Eq. (C.32) we can upper bound Eq. (C.31) by
\[
\Gamma \left( \frac{3}{\alpha} + \frac{m-2}{\alpha} \right) \Gamma \left( \frac{1}{\alpha} \right) \leq \frac{\frac{3}{\alpha} + \frac{3}{\alpha} + 1}{\frac{\alpha}{\alpha} + 1} \left( \frac{\frac{3}{\alpha} + \frac{3}{\alpha} + 2}{\frac{\alpha}{\alpha} + 2} \right) \cdots \left( \frac{\frac{3}{\alpha} + \frac{3}{\alpha} + \left\lfloor \frac{m-2}{\alpha} \right\rfloor - 1}{\frac{\alpha}{\alpha} + \left\lfloor \frac{m-2}{\alpha} \right\rfloor - 1} \right)
\]
\[
= \prod_{k=0}^{\left\lfloor \frac{m-2}{\alpha} \right\rfloor} \frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k},
\]
Using
\[
\frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k} \leq \frac{\alpha}{2} + \frac{k}{2} = \frac{3 + 2k}{1 + 2k^2}, \quad \alpha > 2, \quad k \geq 0,
\]
with equality if and only if \( k = 0 \), and
\[
\left\lceil \frac{m-2}{\alpha} \right\rceil \leq \left\lceil \frac{m}{2} - 1 \right\rceil = \frac{m}{2} - 1, \quad \alpha > 2,
\]
we can write
\[
\frac{E[|\hat{x}|^m]}{E[|\hat{x}|^{m-2}]} \leq \prod_{k=0}^{\left\lfloor \frac{m-2}{\alpha} \right\rfloor} \frac{\frac{3}{\alpha} + k}{\frac{1}{\alpha} + k} \leq \prod_{k=0}^{\left\lfloor \frac{m}{2} - 2 \right\rfloor} \frac{3 + 2k}{1 + 2k} = m - 1.
\]
This completes the proof of Eq. (2.45). \(\square\)

### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGC</td>
<td>automatic gain control</td>
</tr>
<tr>
<td>ANN</td>
<td>artificial neural network</td>
</tr>
<tr>
<td>ASK</td>
<td>amplitude shift keying</td>
</tr>
<tr>
<td>ATSC</td>
<td>Advanced Television Systems Committee</td>
</tr>
<tr>
<td>AWGN</td>
<td>additive white Gaussian noise</td>
</tr>
<tr>
<td>BD</td>
<td>blind deconvolution</td>
</tr>
<tr>
<td>BE</td>
<td>blind equalization</td>
</tr>
<tr>
<td>BER</td>
<td>bit error rate</td>
</tr>
<tr>
<td>BG</td>
<td>Benveniste-Goursat</td>
</tr>
<tr>
<td>BGR</td>
<td>Benveniste-Goursat-Ruget</td>
</tr>
<tr>
<td>BPSK</td>
<td>binary phase shift keying</td>
</tr>
<tr>
<td>BSS</td>
<td>blind signal separation (also: blind source separation)</td>
</tr>
<tr>
<td>cdf</td>
<td>cumulative density function</td>
</tr>
<tr>
<td>CDMA</td>
<td>code division multiple access</td>
</tr>
<tr>
<td>CM</td>
<td>constant modulus</td>
</tr>
<tr>
<td>CMA</td>
<td>constant-modulus algorithm</td>
</tr>
<tr>
<td>dd</td>
<td>decision directed</td>
</tr>
<tr>
<td>DFT</td>
<td>discrete Fourier transform</td>
</tr>
<tr>
<td>DPSK</td>
<td>differential phase shift keying</td>
</tr>
<tr>
<td>DQPSK</td>
<td>differential quaternary phase shift keying</td>
</tr>
<tr>
<td>DSP</td>
<td>digital signal processor</td>
</tr>
<tr>
<td>DTV</td>
<td>digital television</td>
</tr>
<tr>
<td>DVS</td>
<td>discrete-valued signal</td>
</tr>
<tr>
<td>EASI</td>
<td>equivariant adaptive separation via independence</td>
</tr>
<tr>
<td>FDMA</td>
<td>frequency division multiple access</td>
</tr>
<tr>
<td>FFT</td>
<td>fast Fourier transform</td>
</tr>
<tr>
<td>FIR</td>
<td>finite impulse response</td>
</tr>
<tr>
<td>FS</td>
<td>fractionally spaced</td>
</tr>
<tr>
<td>FSE</td>
<td>fractionally spaced equalizer</td>
</tr>
<tr>
<td>Abbreviations</td>
<td>Terminology</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>GMSK</td>
<td>Gaussian minimum shift keying</td>
</tr>
<tr>
<td>GSM</td>
<td>global system for mobile communications(^1)</td>
</tr>
<tr>
<td>HDTV</td>
<td>high-definition television</td>
</tr>
<tr>
<td>HO</td>
<td>higher order</td>
</tr>
<tr>
<td>HOS</td>
<td>higher-order statistics</td>
</tr>
<tr>
<td>ICA</td>
<td>independent component analysis</td>
</tr>
<tr>
<td>ICI</td>
<td>interchannel interference</td>
</tr>
<tr>
<td>IF</td>
<td>intermediate frequency</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>independent and identically distributed</td>
</tr>
<tr>
<td>INCA</td>
<td>independent components analysis (predecessor name of ICA)</td>
</tr>
<tr>
<td>IQ</td>
<td>in-phase/quadrature</td>
</tr>
<tr>
<td>ISI</td>
<td>intersymbol interference</td>
</tr>
<tr>
<td>LHS</td>
<td>left-hand side</td>
</tr>
<tr>
<td>LMS</td>
<td>least mean square</td>
</tr>
<tr>
<td>MC</td>
<td>Monte-Carlo</td>
</tr>
<tr>
<td>MCBD</td>
<td>multichannel blind deconvolution</td>
</tr>
<tr>
<td>MI</td>
<td>mutual information</td>
</tr>
<tr>
<td>MIMO</td>
<td>multiple in—multiple out</td>
</tr>
<tr>
<td>ML</td>
<td>maximum likelihood</td>
</tr>
<tr>
<td>MMSE</td>
<td>minimum mean square error</td>
</tr>
<tr>
<td>MSB</td>
<td>most significant bit</td>
</tr>
<tr>
<td>MSE</td>
<td>mean square error</td>
</tr>
<tr>
<td>MU-CMA</td>
<td>multiuser constant modulus algorithm</td>
</tr>
<tr>
<td>NL</td>
<td>nonlinearity</td>
</tr>
<tr>
<td>ODE</td>
<td>ordinary differential equation</td>
</tr>
<tr>
<td>PAM</td>
<td>pulse amplitude modulation (also called ASK)</td>
</tr>
<tr>
<td>PCA</td>
<td>principal component analysis</td>
</tr>
<tr>
<td>pdf</td>
<td>probability density function</td>
</tr>
<tr>
<td>PLL</td>
<td>phase-locked loop</td>
</tr>
<tr>
<td>pmf</td>
<td>probability mass function</td>
</tr>
<tr>
<td>PSK</td>
<td>phase shift keying</td>
</tr>
<tr>
<td>QAM</td>
<td>quadrature amplitude modulation</td>
</tr>
<tr>
<td>QPSK</td>
<td>quaternary phase shift keying</td>
</tr>
<tr>
<td>RF</td>
<td>radio frequency</td>
</tr>
<tr>
<td>RHS</td>
<td>right-hand side</td>
</tr>
<tr>
<td>RLS</td>
<td>recursive least square</td>
</tr>
<tr>
<td>RV</td>
<td>random variable</td>
</tr>
<tr>
<td>SDMA</td>
<td>space division multiple access</td>
</tr>
<tr>
<td>SER</td>
<td>symbol error rate</td>
</tr>
</tbody>
</table>

\(^1\)originally: fr groupe spéciale mobile
List of Symbols

**Latin symbols**

- \( a \): scaling factor, element of \( A \), auxiliary variable
- \( a \): vector of scaling factors
- \( A \): amplitude, random variable
- \( A \): mixing matrix (BSS), matrix defining ODE (BD)
- \( b \): element of \( B \), auxiliary variable
- \( B \): random variable
- \( B \): diagonal matrix of derivatives of NL
- \( c \): auxiliary variable
- \( C \): Euler’s constant \( C = \lim_{n \to \infty} \left( 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n} - \ln n \right) = 0.5772 \ldots \)
- \( c_x^{(n)} \): \( n \)th-order cumulant of \( x \)
- \( d \): delay
- \( D(\cdot) \): Kullback-Leibler divergence
- \( e \): element of \( e \)
- \( e \): vector of error signals
- \( E_S \): signal energy
- \( E \{ \cdot \} \): expectation operator
- \( f(\cdot) \): general function, nonlinearity
- \( g(\cdot) \): nonlinearity, element of \( g \)
- \( g \): vector of nonlinearities
- \( G \): nonlinear expression of \( x \)
- \( h \): filter taps, channel filter, channel vector
- \( (\cdot)^H \): Hermitian operator
- \( H(\cdot) \): entropy function
List of Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H$</td>
<td>echo canceler matrix, channel matrix</td>
</tr>
<tr>
<td>$i$</td>
<td>index</td>
</tr>
<tr>
<td>$I(\cdot)$</td>
<td>mutual information</td>
</tr>
<tr>
<td>$\Re$</td>
<td>imaginary part designator</td>
</tr>
<tr>
<td>$I$</td>
<td>unity matrix</td>
</tr>
<tr>
<td>$j$</td>
<td>$\sqrt{-1}$ (imaginary unit)</td>
</tr>
<tr>
<td>$J$</td>
<td>general cost function, Jacobian</td>
</tr>
<tr>
<td>$k$</td>
<td>weighting factor, index</td>
</tr>
<tr>
<td>$K$</td>
<td>number of terms in nonlinearity, constant</td>
</tr>
<tr>
<td>$K(\cdot)$</td>
<td>cumulant-generating functions</td>
</tr>
<tr>
<td>$K$</td>
<td>diagonal matrix of signs</td>
</tr>
<tr>
<td>$L$</td>
<td>block length, log-likelihood</td>
</tr>
<tr>
<td>$m$</td>
<td>exponent, index</td>
</tr>
<tr>
<td>$M$</td>
<td>number of constellation points</td>
</tr>
<tr>
<td>$M_s$</td>
<td>number of source signals</td>
</tr>
<tr>
<td>$\mu$</td>
<td>$\mu_i$th-order moment of $x$</td>
</tr>
<tr>
<td>$n$</td>
<td>element of $n$, index</td>
</tr>
<tr>
<td>$n_0$</td>
<td>noise power density</td>
</tr>
<tr>
<td>$N$</td>
<td>Gaussian distribution</td>
</tr>
<tr>
<td>$p$</td>
<td>element of $P$, exponent, parameter of ternary distribution</td>
</tr>
<tr>
<td>$p(\cdot)$</td>
<td>probability density function (pdf)</td>
</tr>
<tr>
<td>$P_s$</td>
<td>symbol-error rate</td>
</tr>
<tr>
<td>$P$</td>
<td>global response matrix, permutation matrix</td>
</tr>
<tr>
<td>$Pr(\cdot)$</td>
<td>probability</td>
</tr>
<tr>
<td>$q$</td>
<td>exponent</td>
</tr>
<tr>
<td>$Q(\cdot)$</td>
<td>$Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left(-\frac{t^2}{2}\right) dt$, quantizer function</td>
</tr>
<tr>
<td>$r$</td>
<td>received signal</td>
</tr>
<tr>
<td>$R_p$</td>
<td>Godard parameter</td>
</tr>
<tr>
<td>$\Re$</td>
<td>real part designator</td>
</tr>
<tr>
<td>$R$</td>
<td>cross-correlation matrix</td>
</tr>
<tr>
<td>$s$</td>
<td>element of $s$</td>
</tr>
<tr>
<td>$s$</td>
<td>vector of original source signals</td>
</tr>
<tr>
<td>$t$</td>
<td>time index, Fourier domain of $x$, Taylor series coefficient</td>
</tr>
<tr>
<td>$(\cdot)^T$</td>
<td>transpose operator</td>
</tr>
<tr>
<td>$T$</td>
<td>symbol length, matrix of time delays, threshold of dd phase alg.</td>
</tr>
<tr>
<td>$T$</td>
<td>matrix of time delays</td>
</tr>
<tr>
<td>$u$</td>
<td>element of $u$</td>
</tr>
<tr>
<td>$u$</td>
<td>vector of separated signals</td>
</tr>
<tr>
<td>$v$</td>
<td>vector of virtual sensor signals</td>
</tr>
<tr>
<td>$w$</td>
<td>element of $W$, element of $w$, filter coefficient (weight)</td>
</tr>
<tr>
<td>$w$</td>
<td>equalizer tap vector</td>
</tr>
<tr>
<td>$W$</td>
<td>separation matrix</td>
</tr>
<tr>
<td>$x$</td>
<td>random variable, element of $x$</td>
</tr>
<tr>
<td>$y$</td>
<td>vector of mixed signals</td>
</tr>
<tr>
<td>$y$</td>
<td>random variable, element of $y$</td>
</tr>
<tr>
<td>$y$</td>
<td>output of nonlinearity vector</td>
</tr>
<tr>
<td>$z$</td>
<td>transform domain variable</td>
</tr>
</tbody>
</table>

Greek symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha, \beta$</td>
<td>parameters of generalized Gaussian, two-tailed Gamma, and symmetric Beta distribution</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>dispersion coefficient, nonlinear moments</td>
</tr>
<tr>
<td>$\gamma_{\pm}$</td>
<td>nonlinear moments</td>
</tr>
<tr>
<td>$\Gamma(\cdot)$</td>
<td>Gamma function (generalized factorial)</td>
</tr>
<tr>
<td>$\delta(\cdot)$</td>
<td>Dirac distribution</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>perturbation</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>nonlinear moments</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>singular value, Lagrange multiplier, forgetting factor</td>
</tr>
<tr>
<td>$\kappa_4$</td>
<td>kurtosis</td>
</tr>
<tr>
<td>$\kappa_{\pm}$</td>
<td>eigenvalues of Hessian</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>vector of kurtoses</td>
</tr>
<tr>
<td>$\mu$</td>
<td>step size</td>
</tr>
<tr>
<td>$\xi$</td>
<td>performance measure, auxiliary variable</td>
</tr>
<tr>
<td>$\rho$</td>
<td>roll-off factor of raised-cosine filter</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>variance</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>vector of variances</td>
</tr>
<tr>
<td>$\tau$</td>
<td>timing offset, time delay</td>
</tr>
<tr>
<td>$\vartheta$</td>
<td>threshold parameter</td>
</tr>
<tr>
<td>$\theta$</td>
<td>vector of threshold values</td>
</tr>
<tr>
<td>$\psi$</td>
<td>phase offset</td>
</tr>
<tr>
<td>$\phi(\cdot)$</td>
<td>characteristic function</td>
</tr>
<tr>
<td>$\chi(\cdot)$</td>
<td>condition number</td>
</tr>
<tr>
<td>$\omega$</td>
<td>frequency</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>normalized frequency</td>
</tr>
</tbody>
</table>
Other symbols

* convolution
(·)∗ conjugate-complex operator
⊙ element-wise multiplication (Schur or Hadamard product)
∝ proportional to
1 vector of ones
# cardinality
! factorial
∥∥ vector norm ∥∥ H 2
· normalized variable (unit variance)
[ ] the integer not greater than
{ } the integer not smaller than

Bibliography


Index

acoustic applications, 1, 42, 44, 58, 140
activation function, see functions
adaptive threshold nonlinearity, 122, 125
adjacent samples, 142
Agee, 3, 143
algorithms
  adaptive threshold nonlinearity, 122, 125
  asymmetric ICA, 48, 93
  Benveniste-Goursat’s, 145, 147
  Benveniste-Goursat-Ruget, 144, 147
  Bussgang, 139, 142, 147
  constant-modulus, 3, 28, 143, 147
decision-directed, 19, 145, 147, 148, 166, 184
  EASI, 57, 93, 179
  extended Bussgang, 149, 162
  extended constant-modulus, 163
  extended Infomax, 93, 120, 128
  extended Sato, 160
  Godard’s, 143, 147
  Infomax, 53
  LMS, 3, 77, 124
  minor component analysis, 131
  natural gradient, 47, 147
  polyspectral, 139
  principal component analysis, 70
  probabilistic, 139
  RLS, 48
  Sato, 142, 147, 157
  Stop-and-Go, 145, 147
symmetric ICA, see EASI
Amari, 4, 43, 47, 56, 71, 74, 147
ambiguities
delay, 141
permutation, 40
phase, 40, 141
    scaling, 40, 75, 141
sign, 40
anti-decay term, 54
anti-Hebbian term, 54
automatic gain control, 75, 124, 179
AWGN channel, 69, 129
baseband representation, 16, 79, 170
beamforming, 44
Bell, 4, 53, 54, 85
Benveniste, 3, 4, 138, 144, 147, 157
Benveniste-Goursat’s algorithm, 145, 147
Benveniste-Goursat-Ruget algorithm, 144, 147
Bernoulli-de l’Hospital’s rule, 36
biomedical engineering, 1
blind deconvolution, see blind equalization
blind equalization, 3, 137–168
  assumptions, 140
  biased, 166
  Bussgang algorithms, 139, 142, 147
  center-tap initialization, 140
  classification, 139
  complexity, 191
  computer simulation, 163

227
## INDEX

convergence, 146, 190  
cost function, 142  
equilibrium, 156  
history, 3  
HOS methods, 139  
iterative methods, 191  
model, 140  
natural gradient, 147, 166  
stationary point, 149  
threshold nonlinearity, 166  

blind estimation of co-channel signals, 44  
blind phase synchronization, 169–188  
8-VSB system, 181  
computer simulation, 185  

blind signal extraction, 58, 71  
blind signal separation, 4, 39–71  
assumptions, 40  
asymmetric ICA, see independent component analysis  
AWGN channel, 129  
biased, 129  
complexity, 191  
computer simulation, 49, 112, 126, 133  
convergence, 49–51, 77, 190  
cost function, 48, 52  
difficulty, 27  
equilibrium, 51  
history, 4  
iterative methods, 71, 191  
maximum-likelihood, 44–47  
model, 40  
overdetermined, 69–71, 131  
partial, 69  
performance, 49  
problem formulation, 40  
QAM signals, 79  
scaling, 40, 66  
stability analysis, 62–65, 101  

stability region, 66–68  
stabilization, 93, 119  
steady-state performance, 77, 78  
symmetric ICA, see EASI  
transpose property, 93  
unbiased, 129  
dertermined, 69  

Bussgang  

algorithms, 139, 142, 147  
extended algorithms, 149, 162  
nonlinearity, 74  
type, 74  
capacity, 28, 140  
Cardoso, 4, 47, 53, 57  
carrier recovery, 80  
Cartwright, 181  

Cauchy-Schwarz inequality, 30, 202, 205  
Čebyšev’s inequality, 204  
cell hand-over, 37  
cellular network, 37  

Central Limit Theorem, 28, 31, 55  
channel fading, 69, 177  
channel vector, 152  

characteristic function, 10  
Chen, 181  
Chevalier, 70  
Cichocki, 71, 121, 127  
clock recovery, 32, 37, 170  
“cocktail-party effect”, 1  

coherent detection, 198  
Comon, 56  
conditional probability density, 45  

constant-modulus algorithm, 3, 28, 143, 147  
extended, 163  
multiuser, 57  
constellations, 19, 64, 109, 177, 183, 198  
contrast function, see cost functions  
convolution, 108  

double, 149  

convolutive noise, 142  

Corollary  
minimum kurtosis, 30  

ratio of Gamma functions, 27  

ratio of HO moments, 22  

universality of scaled threshold nonlinearity, 118  
cost functions, 48, 52, 142  

combination, 148  
quadric, 148  

cost surface, 28  
criteria  

CMA (multiuser), 57  

constrained, 57  

information maximization (Info-max), 53–55  
kurtosis, 58  

maximum-likelihood, 44–47  
maximum-ratio combining, 37  

mutual information, 53  
negentropy, 55  
cross-correlation, 82  
cumulant generating function, 10  
cumulants, 9–12  

composite signal, 31  

definition, 9  

fourth-order, 32, 196  
second-order, 32  
cyclostationarity, 52, 139  

Darlington, 12  

data detection, 139  

data mining, 1  

data reusing, 148  

decision region, 172  

decision-directed methods, 3, 19, 145, 147, 148, 166, 184  
deconvolution, 139  
decorrelation, 3, 50, 52, 74  
demodulation, 37  

Deshpande, 145  
differential equation, 117, 150  
digital signal processor, 126  

discrete Fourier transform, 152  
dispersion constant, 143  

distortion, 42  
distributions  

arbitrary, 119  

bimodal, 14  
class, 88  
complex, 80  

continuous, 20  

crossing points, 117  
cumulative, 54  

δ-, 102, 109, 131  
difficult, 90, 115  
discrete, 16–20, 108, 131  

flat, 83  

Gaussian, 13, 41, 74, 117, 183  
generalized Gaussian, 13, 67, 85, 105  

joint, 52, 56  
lack of shoulder, 12  

Laplacian, 13, 14, 49  

marginal, 56  

mode, 67, 109  

moment ordering, 21–27  
multimodal, 118  

non-Gaussian, 57, 74, 115  
nomodal, 18  

output, 108  

parametric discrete, 19  

parametric, 12, 47  

peaky, 12, 83  

source, 49, 56, 76  

sub-Gaussian, 11, 13, 82, 178  
super-Gaussian, 11, 13, 49, 82  

symmetric, 102  
symmetric Beta, 15, 106  
tail, 12, 118  

two-tailed Gamma, 14, 106  
uniform, 13, 53  

unimodal, 14  
dithering, 134  
diversity reception, 37, 69  

Douglas, 70, 90, 120, 127, 147, 166, 179
EASI algorithm, 57, 93, 179
echo cancellation, 58
eigenvalues, 65, 77, 151, 159
entropy maximization, see information maximization
equalizer
coefficient norm, 157
fractionally spaced, 138
two-sided infinitely long, 150
equilibrium, 108
Euler’s constant, 26
Euler-Lagrange equation, 84
excess noise, 105, 146, 183
extended Infomax algorithm, 93, 120, 128
feedback, 58
fidelity measure, 43
filters
equalization, 139
FIR, 44
ideal lowpass, 35
raised-cosine, 34
Finucan, 28
Fisher efficiency, 74
frequency error, 113
functions, 193–196
activation, 42, 77, 119
concave, 83
convex, 24, 64, 83, 193
cost, 48, 52, 142
cumulant generating, 10
factorial, 193
Gamma, 13, 193
global transfer, 62
monomial, 77
monotonic increasing, 24
nonlinear, see nonlinearities
probability mass, 108, 131
score, 98
tail, 171
Fyfe, 55, 57
Gamma function, 13, 193
recursive property, 26
Garat, 47
Gaussian kernels, 109, 116
Gaussian noise, 37, 38, 69
Gaussianness, 11
Georghiades, 176
Girolami, 55, 57
global stability, 49
global system for mobile communications, 138
Godard, 3, 143, 147
Godard parameter, 143
Godard’s algorithm, 143, 147
Goursat, 3, 147
gradient
complex, 81
modification, 148
natural, 4, 47–49, 148
Newton, 148
nonisotropic magnitude structure, 48
quasi-Newton, 148
relative, 4, 47–49
search method, 46, 55
self-orthogonalizing, 148
standard, 48
stochastic, 47, 123, 140, 146
Gram-Charlier approximations, 56
Grand Alliance, 170
Hadamard product, 132
Hanzo, 19
Hatzmakos, 145
Haykin, 139, 166
Hérault, 4, 71
Hermite operator, 62
Hilbert transform, 170, 173
Hildebrand, 12, 18
homomorphic signal processing, 3
i.i.d. process, 11, 34, 41, 69
Lagrange multiplier, 84
Laheld, 57
Lee, 127, 177, 181
Lemma
composite kurtosis (lower bound), 32, 201
composite kurtosis (upper bound), 31, 199
expectation of a product of two functions, 201
fourth-order moments, 29
ratio of Gamma functions, 25
relationship of HO moments, 22
sufficient stability conditions for blind equalization, 155
leptokurtic, 11
Li, 146
LMS algorithm, 3, 77, 124
local minima, 48
Loève, 22
log-likelihood, 45
Lucky, 3, 147
matrix
block, 61
block notation, 59
channel, 152
covariance equation, 65
characteristic equation, 65
complex scalars, 44
covariance number, 112
cross-correlation, 59
definition, 11
differentiation, 195
element-wise multiplication, 44, 132
Hessian, 148
inversion, 47, 48, 59
invertible, 41
mixing, 40
negative definite, 152
nonsingular, 60, 152, 194
permutation, 41, 43, 50
positive (semi-)definite, 152
scalar, 40
separation, 40, 52, 59
square matrix, 41
time-invariant, 41
Matrix-Inversion Lemma, 48, 194
maximum mean square error, 45, 69, 132, 143
Mbarek, 146
mesokurtic, 11
minimum mean square error, 45, 69, 132, 143
minor component analysis, 131
misadjustment, 77
mixed kurtosis, 128
mixing noise, 111
mixing process, 40
model independence, 57, 85
modem standards, 19
modulation, 177, 198
alphabet size, 170
ASK, 18
BPSK, 109, 143, 198
classification, 28
digital, 79
FSK, 198
PAM, 18, 109, 131, 170, 198
PSK, 198
QAM, 18, 143, 170, 198
SSB, 170
variable-rate star QAM, 19
VSB, 170
moments, 9–12, 196
definition, 9
fractional, 23
Gaussian, 13
generalized Gaussian, 13, 24
Laplacian, 13
nonlinear, 77, 83
ordering, 21–27
ratio, 25
symmetric Beta, 16
two-tailed Gamma, 14
uniform distribution, 13
Monte-Carlo simulations, 169
most significant bit, 143
multichannel blind deconvolution, 2, 71
multiple in—multiple out systems, 2
mutual correlation, 59
mutual independence, 41, 45, 49
mutual information, 53, 56–57
narrowband model, 43–44
negentropy, 55
neural networks
activation functions, 42, 77, 119
degree of freedom, 119
feed-forward, 4, 71
recurrent, 4, 71
single-layer, 42, 126
Nikias, 139
noise resistance, 177
noisy mixture, 45, 129
nonblind adaptation, 59
noncoherent detection, 198
nonlinearities, 73–96
change, 93
cubic, 77, 90, 167, 182
fixed, 90
general, 182
hyperbolic tangent, 49, 90
I/Q-independent, 81
location, 139
monomial, 66, 77
optimal, 82
phase-preserving, 80
polynomial, 85
scaling, 75–82
sigmoid, 77
signum, 102, 106
switch, 120
threshold, 97–136
universal, 85, 115
nonminimum-phase systems, 140
nonstationarity, 52, 148
nuisance parameter, 45
Nyquist spatial sampling, 44
objective function, see cost functions
objective functions
CMA, 57
kurtosis, 57
open-eye condition, 145
orthogonality, 179, 198
orthonormality, 179
outliers, 51
output normalization, 51, 75, 143
Paarmann, 137
Papadias, 57
penalizing factors, 157
perfect-inversion feasibility, 138
performance measure
interchannel interference, 43
intersymbol interference, 142
permutation, 40, 119, 132
perturbation, 62, 150
Pham, 47
phase, 40
phase-error correction, 113, 170
blind, 177
data aided, 177
residual, 176
tracker, 180
phase-locked loop, 170, 177, 185
Picchi, 145, 147, 184
pilot tones, 138
platykurtic, 11
polyspectral algorithms, 139
power transfer, 43
Prati, 145, 147, 184
preprocessing, 70
prewhitening, 58, 157
Price’s theorem, 75
principal component analysis, 70
Proakis, 28, 139
probabilistic algorithms, 139
propagation, 43
quadrature component, 81, 173, 178
quantizer, 145
multiple-level, 100
three-level, 100
raised-cosine filter, 34
roll-off factor, 34, 36
random variables, 9, 49, 66, 76, 201
independent, 31
real-time implementation, 2, 46, 47
receiver, 37
residual mixing, 49, 108
Riemannian structure, 48
RLS algorithm, 48
robustness, 145
same-source lock-on, 43, 58
sample kurtosis, 120
Sato, 3, 142, 147
Sato algorithm, 142, 147, 157
extended, 160
scaling constraint, 83
scatter diagram, 170, 185
Schobben, 58
Schur product, 132
score function, 46, 75, 76
complex distribution, 81
generalized Gaussian, 98
justification, 85
threshold nonlinearity, 100
uniform distribution, 98
second-order statistics, 42, 52, 138
Sejnowski, 4, 53, 54
self-recovering, see blind
semiblind signal separation, 58–62
sensors
displacement, 44
noise, 70
virtual, 60
separation, see blind signal separation
Sgreignoi, 185
Shalvi, 4, 32
sideband, 170
signal-to-noise ratio, 184
estimation, 37
signals
discrete-valued, 146
filtered, 31
reference, 60
temporally correlated, 52
About the Author

Heinz Mathis was born in Zurich in 1968. After visiting primary school in Schlieren / ZH, he attended the Gymnasium in Urdorf, Switzerland, where he received a Matura Typus C. In 1987 he joined the Swiss Federal Institute of Technology Zurich (ETH) to study electrical engineering, with a one-year break in the middle, to study English in Bristol, U.K., and to work as a computer engineer in Silicon Valley. He graduated with a Diploma degree in electrical engineering from ETH in 1993. After a couple of years in industry with Ascom Radiocom in Solothurn, Switzerland, and Philips Paging in Cambridge, U.K., where he was involved in the design of signal processing algorithms for wireless equipment, he returned to ETH, where he started as a teaching assistant at the Signal and Information Processing Laboratory (ISI) in 1997. From 1999 to 2001 he was a research assistant at the same institute, where he got his post-diploma degree in Information Technology [95] in 2000 and his Ph.D. in 2001.