

Master's Thesis

Spatial Transformations for the Alteration of Ambisonic Recordings

Matthias Kronlachner

Graz, June 2014

Institute of Electronic Music and Acoustics

University of Music and Performing Arts, Graz

Graz University of Technology

Advisor:

DI Dr. rer. nat. Franz Zotter

Assessor:

o.Univ.-Prof. Mag. art. DI Dr. techn. Robert Höldrich

Abstract

The production and re-production of sound scenes in the Ambisonic domain offers flexibility regarding the loudspeaker placement around the listening area. Correct decoding should result in a spatial audio perspective that is independent of the loudspeaker configuration. In case a modification of this perspective is needed, or directional alterations of the amplitude, applying such transformations in the Ambisonic domain is more challenging than directly modifying the production objects or loudspeaker placement. Nevertheless, for closing an important gap in flexibility and consistency, making available such algorithms in the Ambisonic domain is beneficial and their development is presented in this work. Most publications are restricted to modifications of first-order Ambisonics. One publication even claims that a dominance transform only works for first order. Nevertheless, this thesis takes the existing approaches, extends them, and makes them available to a broader audience.

Kurzfassung

Die Produktion und Reproduktion akustischer Klangszenen in der Ambisonicsdomäne bietet in erster Linie eine Flexibilisierung in der Lautsprecheraufstellung. Bei korrekter Dekodierung sollten stets perspektivische Eindrücke der ambisonischen Klangszene von der Lautsprecheraufstellung unabhängig werden. Sollte jedoch eine solche perspektivische Verzerrung notwendig werden, oder eine richtungsbezogene Veränderung der Lautstärke, dann stellt dies in der Ambisonicsdomäne eine größere Herausforderung dar als z.B. eine Manipulation der Produktion und ihrer Objekte, der Lautsprecheraufstellung oder der Lautsprechersignale. Zugunsten der Flexibilität und Konsistenz jedenfalls ist die Erarbeitung solcher Algorithmen in der Ambisonicsdomäne dennoch machbar und eine wertvolle Herausforderung, auf die in dieser Arbeit eingegangen wird. Bisherige Publikationen beschränken sich überwiegend auf Transformationen ambisonischer Signale erster Ordnung. Es findet sich sogar die Aussage, der Effekt der 'Dominance' sei für höhere Ordnungen nicht möglich. Die Arbeit greift die dennoch vorhandenen Ansätze auf, erweitert sie, und macht sie einem breiten Publikum anwendbar.

Acknowledgements

I would like to thank my advisor Franz Zotter for guiding me through all stages of this thesis, sharing his knowledge as well as experience and providing me with great opportunities and contacts to apply the outcome of this work.

Many thanks to all my teachers in Graz and the whole staff of the Institute of Electronic Music and Acoustics for the unusual extensive support during the whole period of my studies. Thank you Alois Sontacchi and Brigitte Bergner! Special thanks to Winfried Ritsch who taught me a lot while working on various art projects and IOhannes Zmölnig for guiding me along the first steps of serious audio programming. Thank you Peter Plessas, Thomas Musil, Matthias Frank, David Pirrò, Martin Rumori and Gerriet Sharma for sharing your experience and work with me.

Thanks to my colleagues and friends from Graz especially Marian Weger, Paco Langjahr, Peter Venus and Fabio Kaiser for the fruitful discussions and the numerous concerts and installations we did together!

Thanks to my colleagues from the Music Innovation Studies Centre at the Lithuanian Music and Theatre Academy in Vilnius for their support. Thank you Ričardas Kabelis, Mantautas Krukauskas, Tadas Dailyda, Jurgis Jarašius and Ignas Juzokas!

Last but not least I'd like to thank my family for the support during my whole studies. Without them I would not be able to do what I love. Thank you Laura!

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Place

Date

Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Ort

Datum

Unterschrift

Contents

1	Introduction	1
1.1	Tangible Symbols Describing Continuous Surround Signals	2
1.2	Outline	3
2	Fundamentals: Ambisonics	5
2.1	Coordinate System	5
2.2	Plotting Spherical Datasets/Surround Signals	6
2.3	Spherical Harmonics	8
2.3.1	Ambisonic Conventions	9
2.3.2	Vector Notation for Spherical Harmonics	11
2.3.3	Orthogonality/Orthonormality	11
2.3.4	Spherical Harmonic Transform	11
2.3.5	Discrete Spherical Harmonics Transform	12
2.3.6	Encoding	13
2.3.7	Decoding	13
2.3.8	Computational Efficiency of SH Implementations	14
2.4	Sampling the Sphere	14
2.4.1	Spherical t -designs	15
2.4.2	Finding t -designs	17
3	Transformations of Ambisonic Recordings	21
3.1	Describing any Transformation in the Angular Domain	22
3.1.1	Transformation using t -designs	24
3.2	Rotation in Three Dimensions	25
3.3	Directional Loudness Modifications	27
3.3.1	Spherical Caps	28
3.3.2	Spherical Slepian Functions	32
3.4	Warping	34

3.4.1	Warping Towards a Pole	36
3.4.2	Warping Towards and Away from the Equator	41
3.5	Other Transformation Curves	44
3.6	Transformations through Symmetry Properties	46
4	Practical Contributions	48
4.1	ambiX Audio Plug-ins	49
4.2	mcfx Audio Plug-ins	54
4.3	Metering Ambisonic Signals	56
4.4	Loudspeaker Installations	57
5	Conclusion and Outlook	61
	Bibliography	62

Chapter 1

Introduction

Spatial audio has been a hot research topic for several years. The new generation of cinema surround sound technologies include loudspeakers on different elevation levels thus allow for the playback of full periphonic surround content. Brands like *Dolby*[®] *Atmos*[®] and *Auro 3D*[®] are pushing surround sound with height towards cinemas as well as home users.

In the field of contemporary music composers and sound artists have been experimenting since many years with surround playback facilities that cover the whole area around the listener, e.g. the Philips Pavilion (cf. Fig. 3.18) by Le Corbusier and Iannis Xenakis at the world expo 1958 in Brussels or Karlheinz Stockhausen in the spherical concert hall at the world fair Osaka 1970. Following this idea many research institutions worked on recording, synthesizing and playing back surround sound with various microphone and loudspeaker setups (cf. Fig. 1.1).

One of the first attempts to synthesize an entire sound-field is called Ambisonics and was developed by Michael Gerzon [Ger73] in the 1970s at the University of Oxford. The theory of first-order Ambisonics introduced by Gerzon was extended to higher-order Ambisonics, which allows to create a more distinctive spatial image in the listening area.

While working with spherical surround sound, the necessity gets apparent for modifying (mastering) the spatial image of such recordings. So far the literature lacks a generic description of manipulations for higher-order Ambisonic recordings. This thesis takes existing approaches for first-order Ambisonics, extends them and makes them available to a wide user base.

The algorithms developed are useful for recording and mixing engineers as well as computer musicians working with surround sound. The manipulations can be applied in post processing of spherical microphone array recordings or for correcting the spatial image in different playback venues without having to physically move loudspeakers.

During the work on this thesis, all algorithms have been implemented as audio plugins that can be used in digital audio workstations. Together with the renowned contemporary composers Natasha Barrett, David Monacchi and Gerriet K. Sharma the software has been tested and extended to fit the practical needs.



Figure 1.1: Hemispherical loudspeaker arrangement in the IEM-CUBE, Icosahedral loudspeaker array and mh acoustics Eigenmike[®] represent the motivation for this thesis.

1.1 Tangible Symbols Describing Continuous Surround Signals

In this thesis, I formally use lowercase letters to denote scalars (s, ϕ, ϑ) , lowercase bold letters to denote vectors $(\mathbf{v}, \mathbf{y}, \boldsymbol{\theta})$ and uppercase bold letters for matrices $(\mathbf{M}, \mathbf{Y}, \boldsymbol{\Theta})$. The aim of this thesis is to describe the manipulation of a time- and direction-continuous signal $f(\boldsymbol{\theta}, t)$ as depicted in Fig. 1.2. One conceivable manipulation is to apply a direction-dependent gain to obtain the manipulated signal $\tilde{f}(\boldsymbol{\theta}, t) = g(\boldsymbol{\theta}) f(\boldsymbol{\theta}, t)$. Another useful manipulation is direction reassignment which can be obtained by using an angular mapping function $\tilde{f}(\mathcal{T}\{\boldsymbol{\theta}\}, t) = f(\boldsymbol{\theta}, t)$.

1.2 Outline

Chapter 2 gives an introduction in Ambisonics and the underlying theory using spherical harmonics. The basic mathematical tools are presented and a sampling scheme for the spherical surface is described.

Chapter 3 describes a general approach for performing any transformation on the spherical surface. Furthermore this approach is used to obtain specific transformations such as rotations, directional loudness modifications and warping.

Chapter 4 presents the software implementations and their field of application. An extensive suite of mastering plug-ins for the production of Ambisonic surround sound and general multichannel audio processing is being discussed.

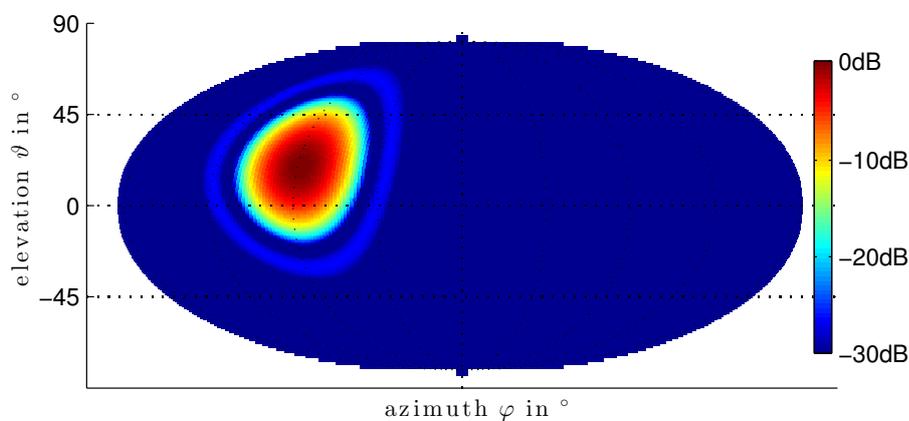


Figure 1.2: Direction-continuous signal.

Chapter 2

Fundamentals: Ambisonics

2.1 Coordinate System

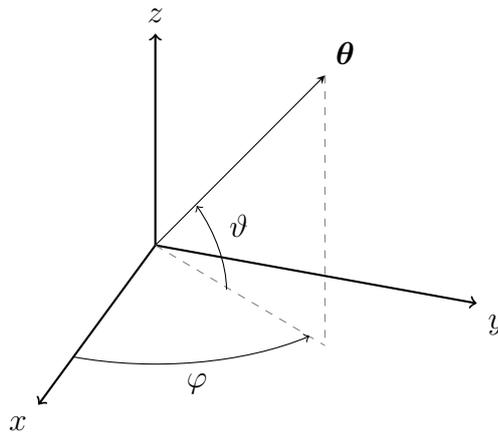


Figure 2.1: Cartesian and spherical coordinate system

The coordinate system used here is defined such that (cf. Fig. 2.1): the x-axis points to the front, the y-axis to the left and the z-axis to the top from the perspective of the audience. Within Ambisonics we mostly deal with spherical coordinates whereby φ is the azimuthal angle in mathematical positive orientation (counter-clockwise)¹ and ϑ being the elevation angle with 0° pointing to the equator and $+90^\circ$ pointing to the north pole.

To denote the directional dependency of the surround signal represented in Ambison-

¹Most user interfaces of spatialization software use a clockwise azimuth.

ics, it will often be necessary to convert between a Cartesian unit direction vector

$$\boldsymbol{\theta} = \begin{pmatrix} \theta_x \\ \theta_y \\ \theta_z \end{pmatrix} = \begin{pmatrix} \cos \varphi \cos \vartheta \\ \sin \varphi \cos \vartheta \\ \sin \vartheta \end{pmatrix} \quad (2.1)$$

and the azimuth and elevation angles (φ, ϑ) of the spherical coordinates

$$\varphi = \arctan \frac{\theta_y}{\theta_x}, \quad \vartheta = \arctan \frac{\theta_z}{\sqrt{\theta_x^2 + \theta_y^2}}. \quad (2.2)$$

2.2 Plotting Spherical Datasets/Surround Signals

Surround signals can be understood as time-varying data distributed on the surface of a sphere. Bernschütz [Ber12] investigated the use of map projection from cartography for spherical measurement data. A popular way to represent circular datasets is the polar diagram (cf. Fig. 2.2(d), 2.2(e)). For representing a spherical dataset, the three-dimensional balloon plot (cf. Fig. 2.2(a)) can be equivalently used, which allows to represent phase in color². Within a two-dimensional print medium or screen the balloon plot only represents the data from one specific point of view with the possibility of hiding several important aspects on the opposite side.

For this thesis, the Mollweide map projection (cf. Fig. 2.2(c)) was chosen as it does not distort levels of constant elevation and the full spherical measurement data can displayed within a two-dimensional plot. This thesis uses polar plots in case the spherical data is rotational symmetric and no information is lost due to this representation.

²Phase can be represented as color in polar plots as well.

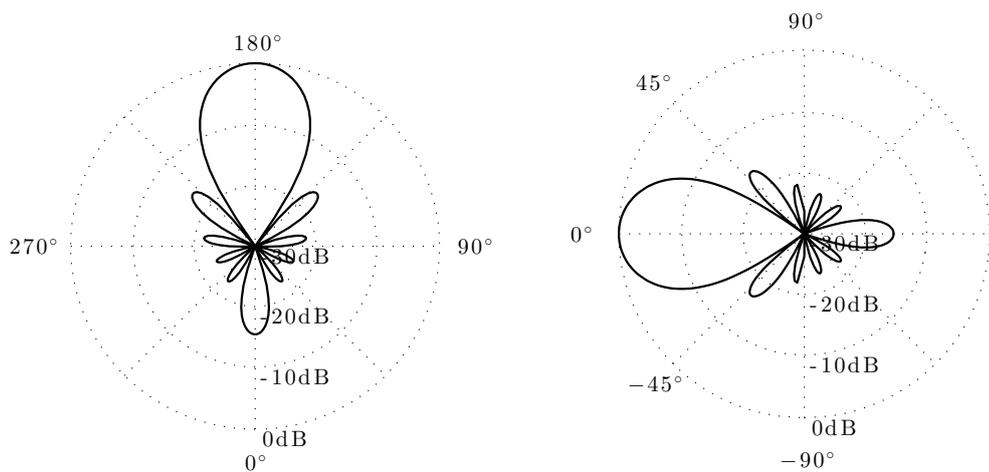
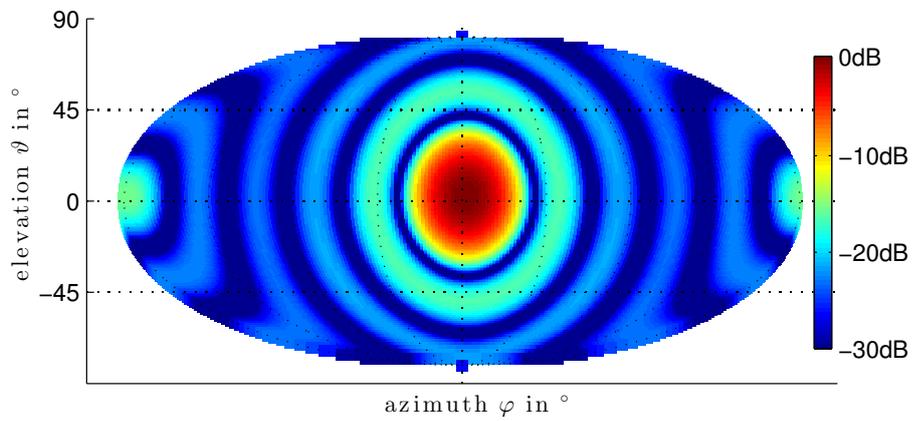
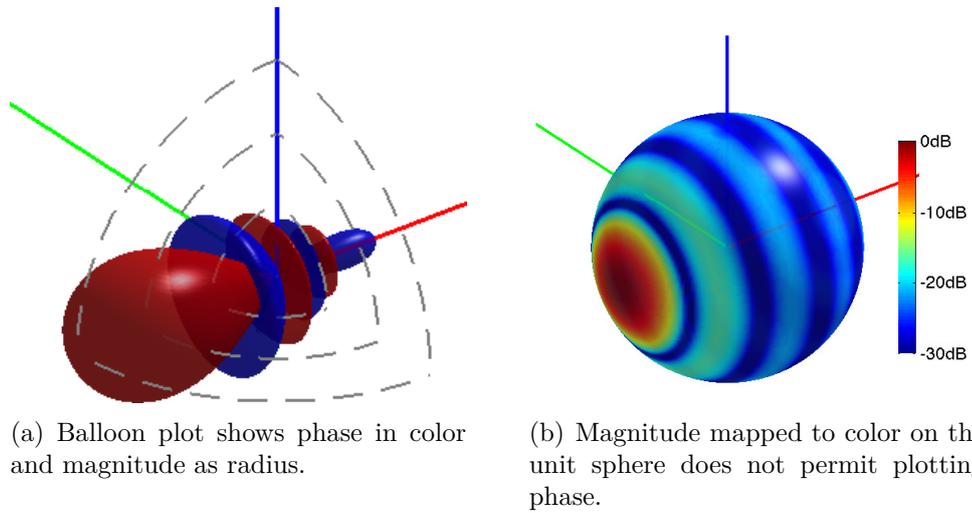


Figure 2.2: Different representations for spherical data. Order $N = 5$ encoded signal with azimuth $\phi = 180^\circ$ and elevation $\theta = 0^\circ$.

2.3 Spherical Harmonics

Spherical harmonics (Fig. 2.3) form a set of orthogonal basis functions and can be used to describe any function on the surface of a sphere [Wil99]. The derivation of spherical harmonics is well covered in the literature [Zm02, Son03, Zot09b].

We consider an audio signal $f(t)$ which arrives from a certain direction $\boldsymbol{\theta} = (\varphi, \vartheta)$. To represent the surround audio signal $f(\varphi, \vartheta, t)$, Ambisonics uses a spherical harmonic expansion up to a truncation order N

$$f(\varphi, \vartheta, t) = \sum_{n=0}^N \sum_{m=-n}^n Y_n^m(\varphi, \vartheta) \phi_{nm}(t) \quad (2.3)$$

whereby Y_n^m being the spherical harmonics of order n , degree³ m and $\phi_{nm}(t)$ the expansion coefficients. With increasing order N the expansion results in a more precise spatial representation.

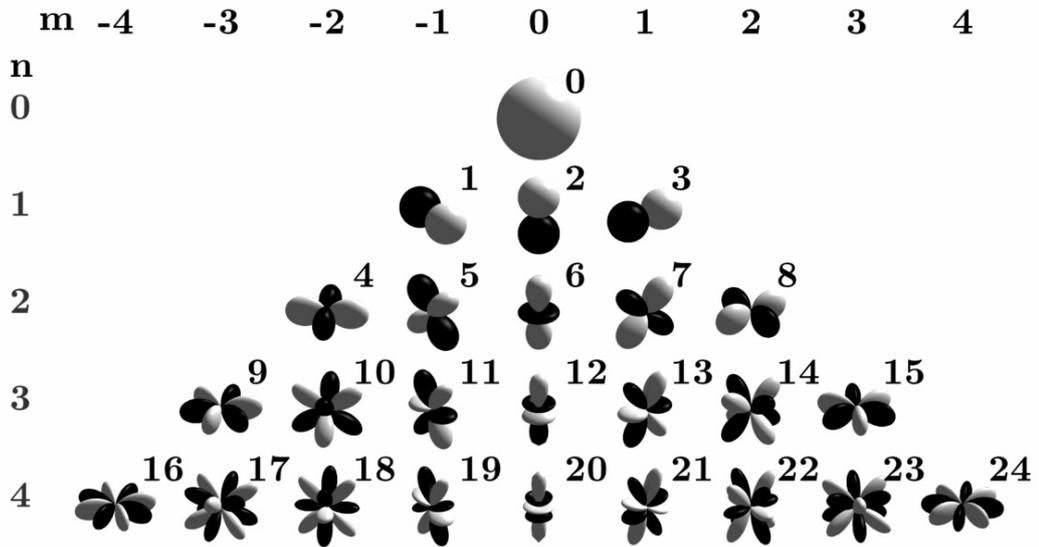


Figure 2.3: Spherical harmonics up to order $N = 4$ with Ambisonic Channel Numbering (ACN) index, order n and degree m .

Spherical harmonics are composed of a normalization term $N_n^{|m|}$, the associated Legendre function $P_n^{|m|}$ and the trigonometric function,

³Several publications use the term order and degree in the opposite way.

$$Y_n^m(\varphi, \vartheta) = N_n^{|m|} P_n^{|m|}(\sin(\vartheta)) \begin{cases} \sin |m|\varphi, & \text{for } m < 0 \\ \cos |m|\varphi, & \text{for } m \geq 0. \end{cases} \quad (2.4)$$

The individual terms of Y_n^m can be efficiently numerically computed by applying the recurrence relations described in [Zot09b].

2.3.1 Ambisonic Conventions

Mathematical elegance as well as practical implementation make it necessary to define a sequence for the expansion coefficients $\phi_{nm}(t)$ in Eq. (2.3). Further the normalization term $N_n^{|m|}$ has to be defined in order to ensure the interoperability of Ambisonic software and the correct playback of Ambisonic recordings.

The history of Ambisonics brought up different conventions regarding the sequence of Y_n^m and the normalization term $N_n^{|m|}$. This thesis uses Ambisonic Channel Numbering (ACN) and, due to mathematical elegance, fully normalized N3D spherical harmonics.

The Ambisonic Channel Numbering defines a sequence for the spherical harmonics as

$$ACN = n^2 + n + m. \quad (2.5)$$

This sequence can easily be deciphered and we get the spherical harmonic order n with

$$n = \lfloor \sqrt{ACN} \rfloor, \quad (2.6)$$

and the degree m with

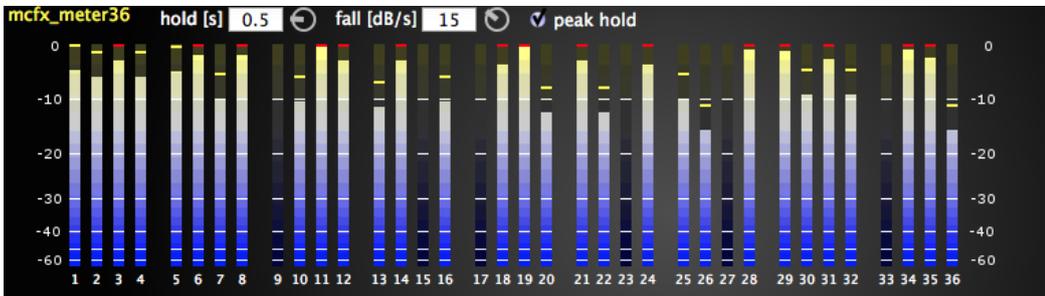
$$m = ACN - n^2 - n. \quad (2.7)$$

We obtain the fully normalized (N3D) spherical harmonics by applying the normalization term as

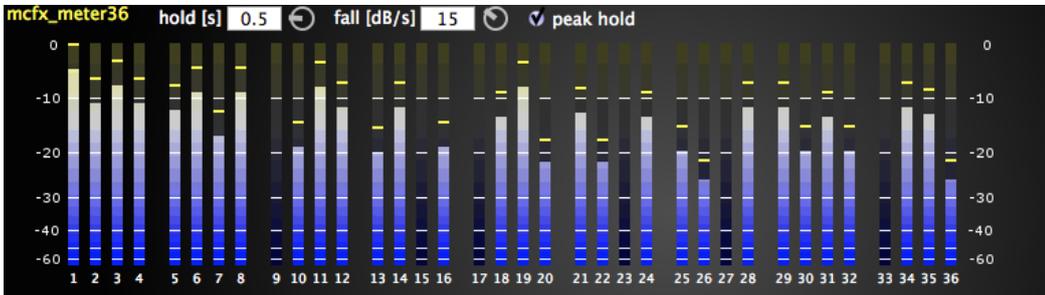
$$N_n^{|m|} = \sqrt{\frac{(2n+1)(2-\delta_m)(n-|m|)!}{4\pi(n+|m|)!}}. \quad (2.8)$$

It is worth mentioning that the *software* (cf. Sec. 4) accompanying this thesis does use Schmidt seminormalized ($N_{SN3D} = \sqrt{(2-\delta_m)\frac{(n-|m|)!}{(n+|m|)!}}$) spherical harmonics as

proposed in ambiX [NZDS11]. While N3D normalization does yield to a set of orthonormal basis vectors and therefore has mathematical advantages, SN3D does have advantages in the exchange of Ambisonic reorderings. Higher-order coefficients of SN3D signal sets do not exceed the level of the zeroth order coefficient ϕ_{00} , while higher-order coefficients of N3D signals are scaled by $\sqrt{2n+1}$ and therefore might exceed ϕ_{00} , which could cause distortion (cf. Fig 2.4) for integer type audio formats. Although Chapman et. al. [Cha09a] give good arguments for the use of N3D as standard format.



(a) N3D normalization might cause distortions in higher orders.



(b) SN3D normalization - no signal component is exceeding ϕ_{00} (level meter 1).

Figure 2.4: Spherical harmonic expansion coefficients level for 0dB peak signal encoded to the direction $\phi = 45^\circ, \vartheta = 45^\circ$ in 5th order using N3D and SN3D normalization. The multichannel metering plug-in depicted here is part of the *mcfx* plug-in suite (cf. Sec. 4.2).

2.3.2 Vector Notation for Spherical Harmonics

Using the Ambisonic Channel Numbering from Eq. (2.5) allows us to define a sequence for the spherical harmonics $Y_n^m(\varphi, \vartheta) = Y_{ACN}(\varphi, \vartheta)$ and the expansion coefficients (a.k.a. Ambisonic signals) $\phi_{nm}(t) = \phi_{ACN}(t)$ from Eq. (2.3) can be stacked into a vector with length $(N + 1)^2$, accordingly,

$$\mathbf{y}_N(\boldsymbol{\theta}) = \begin{pmatrix} Y_0(\boldsymbol{\theta}) \\ Y_1(\boldsymbol{\theta}) \\ Y_2(\boldsymbol{\theta}) \\ Y_3(\boldsymbol{\theta}) \\ Y_4(\boldsymbol{\theta}) \\ \vdots \\ Y_{(N+1)^2-1}(\boldsymbol{\theta}) \end{pmatrix} = \begin{pmatrix} Y_0^0(\boldsymbol{\theta}) \\ Y_1^{-1}(\boldsymbol{\theta}) \\ Y_1^0(\boldsymbol{\theta}) \\ Y_1^1(\boldsymbol{\theta}) \\ Y_2^{-2}(\boldsymbol{\theta}) \\ \vdots \\ Y_N^M(\boldsymbol{\theta}) \end{pmatrix}, \quad \boldsymbol{\phi}_N(t) = \begin{pmatrix} \phi_0(t) \\ \phi_1(t) \\ \phi_2(t) \\ \vdots \\ \phi_{(N+1)^2-1}(t) \end{pmatrix}. \quad (2.9)$$

The angular dependency above is symbolically condensed by using the unit Cartesian direction vector $\boldsymbol{\theta}$ instead of (φ, ϑ) . In the vector notation, an Ambisonic surround signal as in Eq. (2.3) can be written as

$$f(\boldsymbol{\theta}, t) = \mathbf{y}_N^T(\boldsymbol{\theta}) \boldsymbol{\phi}_N(t). \quad (2.10)$$

2.3.3 Orthogonality/Orthonormality

The set of spherical harmonics used here form an orthogonal and due to the normalization term (cf. Eq. (2.8)) an orthonormal set of basis functions.

Therefore the integral along the surface area of the unit sphere \mathbb{S}^2 of the spherical harmonic vector times its transpose results in the identity matrix

$$\int_{\mathbb{S}^2} \mathbf{y}_N(\boldsymbol{\theta}) \mathbf{y}_N^T(\boldsymbol{\theta}) d\boldsymbol{\theta} = \mathbf{I}. \quad (2.11)$$

2.3.4 Spherical Harmonic Transform

Similar to the Fourier transform for decomposing a time-domain signal using orthogonal basis functions into its frequency spectrum, it is possible to decompose a

band-limited function on the unit sphere $f(\boldsymbol{\theta})$ into the spherical spectrum $\boldsymbol{\phi}_N$ using spherical harmonics transform (\mathcal{SHT}) [DH94, Raf05],

$$\mathcal{SHT}\{f(\boldsymbol{\theta})\} = \boldsymbol{\phi}_N = \int_{\mathbb{S}^2} \mathbf{y}_N(\boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta}. \quad (2.12)$$

2.3.5 Discrete Spherical Harmonics Transform

In order to numerically calculate the integral in Eq. (2.12), we can use a suitable distribution of L directions

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_L]^T. \quad (2.13)$$

The choice of sampling points will be discussed in the following section (cf. Sec. 2.4). Using the finite number of directions, we perform the discrete spherical harmonics transform (\mathcal{DSHT}) to obtain the discrete spherical spectrum $\boldsymbol{\phi}_N$

$$\mathcal{DSHT}\{\mathbf{f}(\boldsymbol{\Theta})\} = \boldsymbol{\phi}_N = \mathbf{Y}_N^\dagger(\boldsymbol{\Theta}) \mathbf{f}(\boldsymbol{\Theta}), \quad (2.14)$$

whereby \dagger denotes the Moore–Penrose pseudoinverse

$$\mathbf{Y}^\dagger = (\mathbf{Y}^T \mathbf{Y})^{-1} \mathbf{Y}^T. \quad (2.15)$$

The \mathcal{DSHT} requires to evaluate the $(N + 1)^2$ spherical harmonics $Y_{ACN}(\boldsymbol{\theta})$ for $L \geq (N + 1)^2$ directions (samples). The spherical harmonic vectors form a new matrix $\mathbf{Y}_N(\boldsymbol{\Theta})$ with the dimensions $L \times (N + 1)^2$

$$\mathbf{Y}_N(\boldsymbol{\Theta}) = \begin{pmatrix} \mathbf{y}_N^T(\boldsymbol{\theta}_1) \\ \mathbf{y}_N^T(\boldsymbol{\theta}_2) \\ \vdots \\ \mathbf{y}_N^T(\boldsymbol{\theta}_L) \end{pmatrix} = \begin{pmatrix} Y_0(\boldsymbol{\theta}_1) & Y_1(\boldsymbol{\theta}_1) & Y_2(\boldsymbol{\theta}_1) & \cdots & Y_{(N+1)^2-1}(\boldsymbol{\theta}_1) \\ Y_0(\boldsymbol{\theta}_2) & Y_1(\boldsymbol{\theta}_2) & Y_2(\boldsymbol{\theta}_2) & \cdots & Y_{(N+1)^2-1}(\boldsymbol{\theta}_2) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ Y_0(\boldsymbol{\theta}_L) & Y_1(\boldsymbol{\theta}_L) & Y_2(\boldsymbol{\theta}_L) & \cdots & Y_{(N+1)^2-1}(\boldsymbol{\theta}_L) \end{pmatrix}. \quad (2.16)$$

2.3.6 Encoding

For encoding the position of k sound sources $s_i(t)$ in Ambisonics, we use

$$\boldsymbol{\phi}_N(t) = \sum_{i=1}^k \mathbf{y}_N(\boldsymbol{\theta}_i) s_i(t) . \quad (2.17)$$

The transmission and storage of Ambisonic signals gets efficient as soon as the number of sound sources k is larger than the size of the expansion coefficient vector $\boldsymbol{\phi}_N(t)$. The size of $\boldsymbol{\phi}_N(t)$ depends on the truncation order N and yields $(N + 1)^2$.

Ambisonic signals can also be obtained by recording sound scenes with microphone arrays [Raf05, Ple09, Lös13]. Depending on the array geometry, a specific encoding method has to be chosen. Often this requires a multiple-in-multiple-out (MIMO) system, e.g.,

$$\boldsymbol{\phi}_N(t) = \sum_{i=1}^n \mathbf{H}_i * m_i(t) , \quad (2.18)$$

where $m_i(t)$ is the i -th microphone signal, \mathbf{H}_i the i -th filter matrix, and $*$ denotes the convolution operator.

2.3.7 Decoding

In order to play back Ambisonic recordings using loudspeakers, we have to decode the Ambisonic signal. In case the loudspeakers are regularly distributed around the listening area (cf. Sec. 2.4), the i -th loudspeaker signal $l_i(t)$ at position $\boldsymbol{\theta}_i$ can be obtained by sampling the Ambisonic signal $\boldsymbol{\phi}(t)$ using

$$l_i(t) = \mathbf{y}(\boldsymbol{\theta}_i)^T \boldsymbol{\phi}(t) . \quad (2.19)$$

In practice we often deal with non-regular distributed loudspeaker arrays. The sampling decoder from Eq. (2.19) does not produce optimal results for such arrays. Several approaches have been developed to compute well-behaved decoders for irregular loudspeaker arrangement and can be found in [PZ09, HBL12, ZPN12, ZF12, EJZ14].

2.3.8 Computational Efficiency of SH Implementations

For several applications, such as nonlinear optimization problems in Ambisonic decoder design or the search for optimal point distributions on the sphere (cf. Sec. 2.4.1), a large number of spherical harmonic functions have to be evaluated. Therefore it is worth looking into the implementation of spherical harmonics in our numerical computation environment of choice MATLAB⁴. We want to compare the computation time of two exemplary real-valued spherical harmonic implementations,

1. computing the associated Legendre polynomials $P_n^{|m|}$ using recurrence relations from [Zot09b],
2. using the built-in MATLAB function `legendre` for computing $P_n^{|m|}$.

Each implementation was used to evaluate a set of 240 directions $\boldsymbol{\theta}$ for different spherical harmonic orders N . The average computation time for 30 repetitions is noted in Tab. 2.1.

N	T_1	T_2	T_1/T_2
3	5.69 ms	3.29 ms	1.72
5	7.17 ms	3.17 ms	2.25
10	25.4 ms	8.71 ms	2.91
100	3.2 s	0.656 s	4.87

Table 2.1: Average computation time for different implementations of $\mathbf{y}_N(\boldsymbol{\theta})$ for 240 directions $\boldsymbol{\theta}$ and 30 repetitions.

The test was carried out on a MacBook Pro with 2,53 GHz Intel Core 2 Duo and 8GB Ram running MacOS 10.9.2 using MATLAB v.8.1.0.604 (R2013a).

It can be seen that the choice of an efficient implementation can significantly reduce the computation time and e.g. nonlinear optimization problems solved faster. Another implementation using nonequispaced discrete Fourier transform (NDFT) is provided by [KKP09] and was used in [GP11] for finding spherical designs (cf. Sec. 2.4.1).

2.4 Sampling the Sphere

In order to compute the inverse of the matrix \mathbf{Y}_N for \mathcal{DSHT} , a well-behaved distribution of directions Θ has to be used. Rafaely [Raf05] and Zotter [Zot09b, Zot09a]

⁴MATLAB is a registered trademark of The MathWorks, Inc.

describe several sampling strategies for sound field analysis and synthesis regarding the required number of directions L to perform \mathcal{DSHT} .

Hyperinterpolation uses the minimum number of sampling nodes $L = (N + 1)^2$. Other sampling strategies typically require $L > (N + 1)^2$ nodes.

Other sampling schemes include

- Gauss-Legendre
- Equiangular sampling
- Equiangular cylindric
- Spiral points
- HEALPix

We want to have a closer look into spherical t -designs, which offer advantages for the application in this thesis.

2.4.1 Spherical t -designs

Using a spherical t -design, the discrete summation of a spherical polynomial $\mathcal{P}_n(\mu)$ of limited degree $n \leq t$ equals the integral over the unit sphere \mathbb{S}^2 [GP11],

$$\int_{\mathbb{S}^2} \mathcal{P}_n(\mu) d\mu = \frac{4\pi}{L} \sum_{l=1}^L \mathcal{P}_n(\mu_l). \quad (2.20)$$

The usage of t -designs allow \mathcal{DSHT} without any pseudo-inversion but simple transposition for $t \geq 2N$

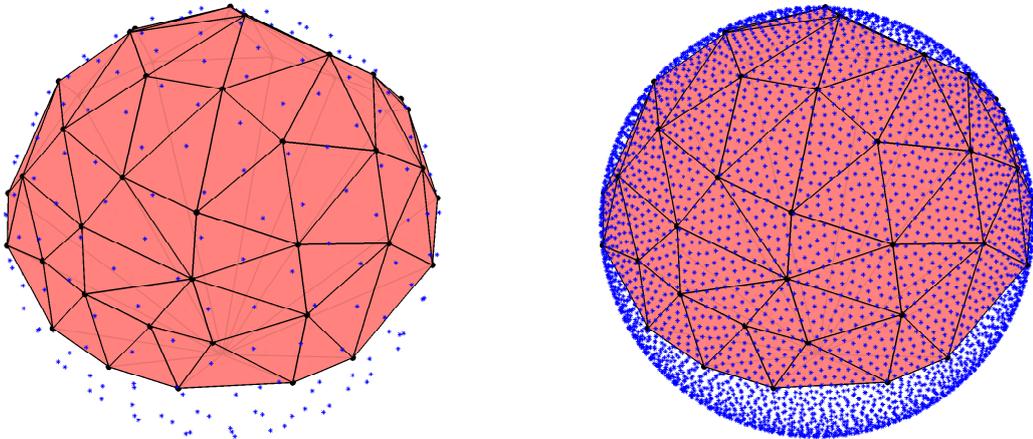
$$\mathbf{Y}_N^\dagger(\Theta_t) = \text{diag}\left\{\frac{4\pi}{L}\right\} \mathbf{Y}_N^T(\Theta_t). \quad (2.21)$$

The Platonic solids are well known examples for spherical t -designs. Hardin and Sloane [HS96] provide⁵ t -designs up to $t = 21$ with 240 nodes. For the derivation of our Ambisonic transformations in the next section, we will use spherical t -designs to numerically compute the integral of two spherical harmonic functions with degree N

⁵<http://neilsloane.com/sphdesigns/dim3/>

each. Therefore, a spherical design with $t \geq 2N$ has to be used. While a 21-design allows the use of up to Ambisonics order $N = 10$ and therefore cover most current Ambisonic encoding/decoding systems there are already applications that require a larger number of nodes L .

The pragmatic *All-Round Ambisonic Decoding* (AllRAD) [ZF12] approach uses a virtual t -design loudspeaker arrangement where the sampling decoder from Eq. (2.19) provides an optimal result. These virtual loudspeaker signals are projected to the real loudspeakers using Vector Base Amplitude Panning (VBAP) [Pul97]. To achieve constant energy about all panning directions, the number of virtual loudspeakers per real loudspeaker triangle should be sufficiently large. Using the 240 nodes 21-design may result in poor interpolation for a high number of real loudspeakers. Fig. 2.5 shows the virtual loudspeaker mapping for computing the AllRAD decoder for the ZKM (Zentrum für Kunst und Medientechnologie) Kubus in Karlsruhe, Germany, which requires a $(2N + 1)$ -design with many nodes for smooth panning about all directions.



(a) $L = 240$, 21-design, several triangles accommodate only one virtual loudspeaker, therefore a larger number of virtual loudspeakers is needed for good interpolation.

(b) $L = 5200$, 100-design fills all triangles with a sufficient number of virtual loudspeakers for interpolation.

Figure 2.5: Comparison of virtual loudspeaker distribution for computing the AllRAD decoder for the hemispherical 43-loudspeaker installation at the ZKM-Kubus. Black dots in the corners of the triangulation indicate the real loudspeaker positions, blue stars represent the virtual t -design loudspeakers used for VBAP interpolation. The figures have been plotted using the Ambisonic Decoder Toolkit by Aaron Heller [HB14].

2.4.2 Finding t -designs

There seems to be no analytic approach for easily finding t -designs of arbitrary orders or number of nodes. However nonlinear optimization can be used to find sampling nodes Θ that fulfill Eq. (2.20). Using a suitable distribution of nodes as initial configuration and a well behaved cost function a minimization problem can be solved to find a t -design. It is necessary to find a good optimization criterion (cost function) $\epsilon(\Theta)$ to allow convergence of the nonlinear solver and to avoid getting stuck in local minima.

Optimization Criteria

1. Condition number κ . The condition number is the ratio between the largest and smallest singular value of a matrix obtained by singular value decomposition [KL80], and it indicates the sensitivity of the system to errors. Ideally the condition number of the spherical harmonic matrix should be equal to one thus allowing proper inversion. Therefore we can formulate a cost function using the condition number as

$$\epsilon_C(\Theta) = \kappa(\mathbf{Y}_N(\Theta)) - 1, \quad (2.22)$$

whereby $N \leq \frac{t}{2}$.

2. Frobenius norm. t -designs fulfill the property

$$\mathbf{Y}_N^T(\Theta_t) \mathbf{Y}_N(\Theta_t) = \text{diag}\left\{\frac{4\pi}{L}\right\}, \quad (2.23)$$

whereby $N \leq \frac{t}{2}$. We can measure the distance of a node configuration Θ from this property using the Frobenius norm. The Frobenius norm of a matrix \mathbf{A} is defined as the square root of the sum of the absolute squares of its elements [Wei14a]

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}. \quad (2.24)$$

We can formulate a cost function indicating the distance to the desired t -design using Eq. (2.23) and the Frobenius norm as

$$\epsilon_F(\Theta) = \|\mathbf{Y}_N^T(\Theta) \mathbf{Y}_N(\Theta)\|_F - \|\text{diag}\left\{\frac{4\pi}{L}\right\}\|_F. \quad (2.25)$$

3. Criterium by Sloan and Womersley. Sloan and Womersley [SW09] presented following characterization which is zero for t -designs and suits as a cost function

$$\epsilon_{SW}(\Theta) = \frac{1}{L^2} \sum_{n=1}^t \sum_{m=-n}^n \left| \sum_{l=1}^L Y_n^m(\theta_l) \right|^2. \quad (2.26)$$

In order to test the suitability of the different criteria, we use a known t -design Θ_t from [HS96], add random noise with increasing standard deviation to each position θ_l and project the new Cartesian vector back onto the unit sphere to get Θ_ν . The average deviation of the noisy t -design from the original Θ_t is obtained by

$$\bar{\nu} = \frac{1}{L} \sum_{l=1}^L \arccos \theta_t^T \theta_\nu. \quad (2.27)$$

Optimization Process

Fig. 2.6 suggests that the criterion from Sloan & Womersley is the best choice for the cost function $\epsilon_{SW}(\Theta)$. We use MATLAB's nonlinear least squares solver `lsqnonlin` to minimize the cost function (2.26) and concatenate the Cartesian coordinate matrix Θ into a row vector which is used as design variable. The solving algorithm may move the Cartesian coordinates in \mathbb{R}^3 . Before evaluating the cost function we have to project Θ back onto the unit sphere. Introducing a threshold for the cost function $|\epsilon_{SW}|$ allows to terminate the optimization within a certain tolerance.

Fig. 2.7(a) shows the residual error for the first 70 of 424 total iterations. The search for a 10-design with 90 nodes took 24 minutes and 36 seconds with tolerance $|\epsilon_{CW}| < 10^{-11}$. The condition number of the starting configuration was 5.56, after optimization the condition number reached $\kappa(\mathbf{Y}(\Theta)) = 1.000008$.

Gräf and Potts [GP11] are using a highly optimized nonlinear optimization method to find t -designs of high order and large number of nodes. They provide⁶ several node configurations for spherical 50-designs with 1300 points up to 1000-designs with 1.002.000 points.

⁶<https://www-user.tu-chemnitz.de/~potts/workgroup/graef/computations/pointsS2>

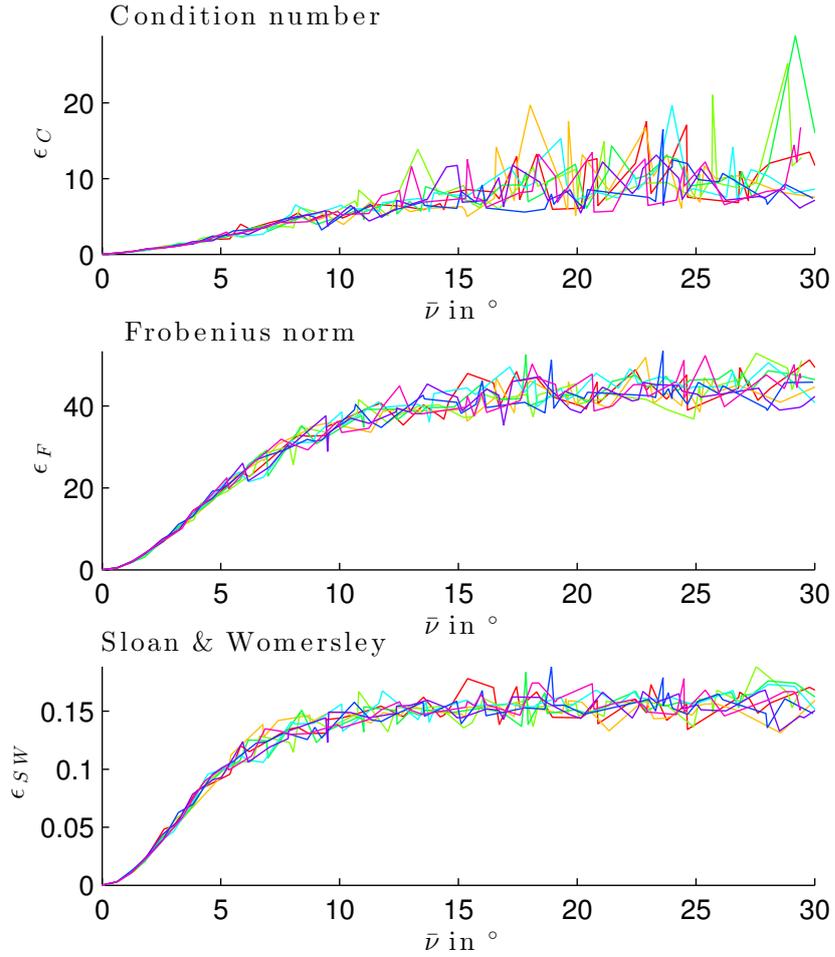
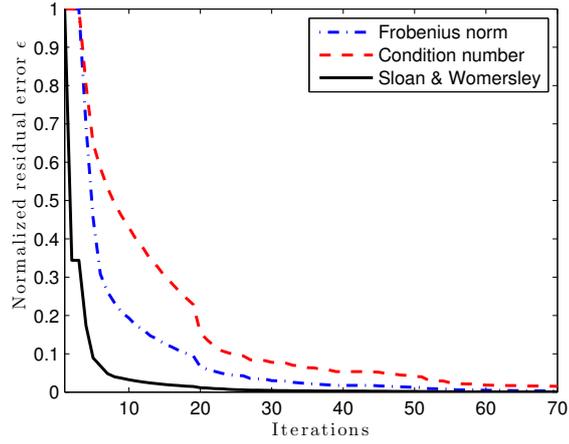
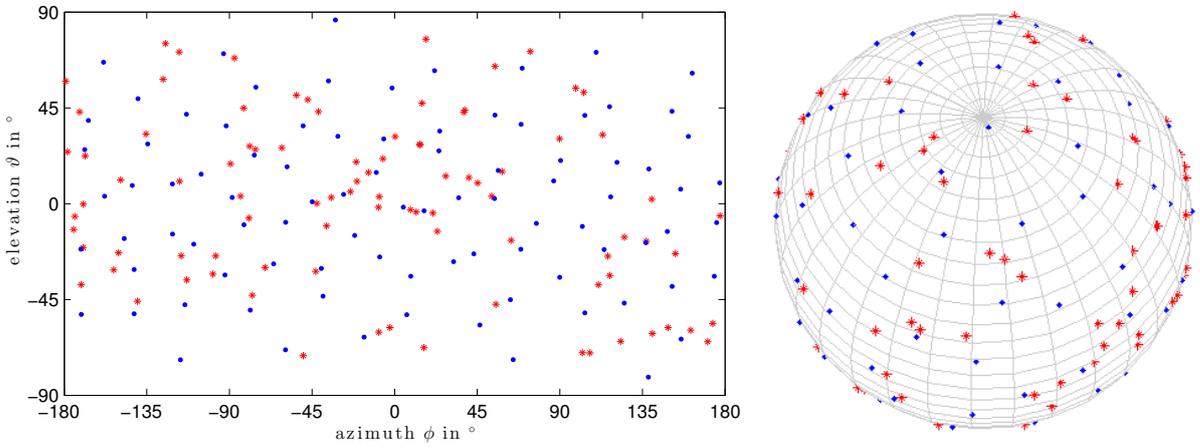


Figure 2.6: Starting from a known 21-design [HS96], random noise is added to the node configuration Θ_t with the average deviation stepwise increased. For each increment, the three proposed optimization criteria are estimated. The plot shows 8 repetitions. The condition number does not offer a well behaved cost function ϵ_C and it is likely that nonlinear optimization will yield in local minima. ϵ_{SW} offers a steeper curve near the optimum than ϵ_F and therefore seems to be the better choice.



(a) First part of the optimization process, ϵ_{SW} was chosen as cost function, ϵ_C and ϵ_F are indicated for comparison only.



(b) Crosses indicate starting configuration, dots indicate the optimized node configuration.

Figure 2.7: Search for $L = 90$, 10–design, using nonlinear optimization and a random start configuration, $|\epsilon_{CW}| < 10^{-11}$, $\kappa(\mathbf{Y}(\Theta)) = 1.000008$.

Chapter 3

Transformations of Ambisonic Recordings

While working with surround recordings, it is desirable to perform spatial transformations. A mixing engineer wants to gain control of the global orientation of the sound scene, of the emphasis of certain directions, or of modifications of the spatial perspective.

Transformations are easily employed whenever the surround recording is available as object-based format. Object-based surround formats deploy each audio source in discrete form and attach meta data, e.g. about their position, width, and loudness. Therefore only the meta data has to be altered in order to perform spatial transformations. Directional Audio Coding (DirAC) [Pul06] can be used to separate sound sources from microphone array recordings and assign meta data regarding their spatial characteristics. The paper [PPP12] describes the process of applying spatial audio effects such as rotation, acoustic zoom (warping) and spatial filtering using DirAC.

Applying transformations of recordings in the Ambisonic domain appear more challenging than modifying object-based audio meta data. Despite the expected efforts to take, a solution would be beneficial and desirable. The separation of sound sources may not always work without audible artifacts, therefore an object-based representation may not always be useful as a practical intermediate representation.

Therefore we search for a way to modify the spatial characteristic of Ambisonically encoded sound scenes.

By simple matrix multiplication, it is possible to apply a frequency and time independent transformation to our Ambisonic signal $\phi_N(t)$, and we obtain the transformed

Ambisonic signal $\tilde{\phi}_{\tilde{N}}(t)$ through

$$\tilde{\phi}_{\tilde{N}}(t) = \mathbf{T} \phi_{\mathbf{N}}(t). \quad (3.1)$$

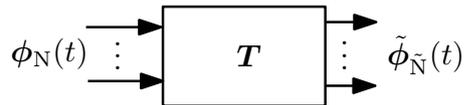


Figure 3.1: Transformation of Ambisonic signals using \mathbf{T} .

Several publications presented ways to obtain transformation matrices \mathbf{T} for various spatial manipulations [PZ11, ZP11, CC09, Cha09b, Son03, Zm02]. However, those either suffer from being limited to certain Ambisonic orders \mathbf{N} or being restricted to one specific spatial manipulation.

The following section describes a more general and pragmatic way to perform transformations numerically in the angular domain. Using the optimal sampling scheme from Sec. 2.4.1 allows to obtain a transformation matrix \mathbf{T} without significant computational complexity for currently in practice used Ambisonic orders \mathbf{N} . Exceeding what is practically useful today, larger \mathbf{N} would require matrix multiplications that are more demanding and an implementation using fast spherical harmonic transform [KKP09] would be preferable then.

In general, the transformed Ambisonic signals might be of higher order $\tilde{\mathbf{N}}$. Using smooth transformation curves helps to keep the re-expansion order $\tilde{\mathbf{N}}$ small.

3.1 Describing any Transformation in the Angular Domain

We want to describe transformations that are using one or both of the following manipulations:

1. weighting by a direction-dependent gain $g(\boldsymbol{\theta})$,
2. applying the angular transformation $\tilde{\boldsymbol{\theta}} = \mathcal{T}\{\boldsymbol{\theta}\}$.

The direction-dependent gain allows to emphasize signals from wanted directions and attenuate them from unwanted directions. Angular transformations change the

direction of arrival for some or all directions, therefore allow to change the spatial perspective.

A surround signal $f(\boldsymbol{\theta}, t)$ weighted by the direction-dependent gain factor $g(\boldsymbol{\theta})$ and mapped to a different direction $\tilde{\boldsymbol{\theta}} = \mathcal{T}\{\boldsymbol{\theta}\}$ results in the modified surround signal $\tilde{f}(\tilde{\boldsymbol{\theta}}, t)$

$$\tilde{f}(\mathcal{T}\{\boldsymbol{\theta}\}, t) = g(\boldsymbol{\theta}) f(\boldsymbol{\theta}, t). \quad (3.2)$$

To simplify later computations of our transformation matrix \mathbf{T} , we will use the inverse angular transformation $\mathcal{T}^{-1}\{\boldsymbol{\theta}\}$ and therefore get

$$\tilde{f}(\boldsymbol{\theta}, t) = g(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) f(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}, t). \quad (3.3)$$

We recall the Ambisonic representation of our surround signal from Eq. (2.10) and insert $f(\boldsymbol{\theta}, t) = \mathbf{y}_N^T(\boldsymbol{\theta}) \boldsymbol{\phi}_N(t)$ into Eq. (3.3)

$$\mathbf{y}_N^T(\boldsymbol{\theta}) \tilde{\boldsymbol{\phi}}_N(t) = g(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) \mathbf{y}_N^T(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) \boldsymbol{\phi}_N(t). \quad (3.4)$$

In order to obtain our transformed Ambisonic signal $\tilde{\boldsymbol{\phi}}_N(t)$, we have to remove $\mathbf{y}_N^T(\boldsymbol{\theta})$ by using the orthogonality after integration over two spherical harmonics (cf. Eq. (2.11)),

$$\tilde{\boldsymbol{\phi}}_N(t) = \underbrace{\int_{\mathbb{S}^2} \mathbf{y}_N(\boldsymbol{\theta}) g(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) \mathbf{y}_N^T(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) d\boldsymbol{\theta}}_{:=\mathbf{T}} \boldsymbol{\phi}_N(t). \quad (3.5)$$

Comparing Eq. (3.5) with Eq. (3.1) gives us the expression for the transformation matrix

$$\mathbf{T} = \int_{\mathbb{S}^2} \mathbf{y}_N(\boldsymbol{\theta}) g(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) \mathbf{y}_N^T(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) d\boldsymbol{\theta}. \quad (3.6)$$

We recognize Eq. (3.6) as spherical harmonics transform (\mathcal{SHT}) (cf. Sec. 2.3.4) over the expression

$$\mathbf{T} = \mathcal{SHT}\{g(\mathcal{T}^{-1}\{\boldsymbol{\theta}\}) \mathbf{y}_N^T(\mathcal{T}^{-1}\{\boldsymbol{\theta}\})\}. \quad (3.7)$$

In order to numerically evaluate the integral from Eq. (3.6), we perform discrete spherical harmonics transform (cf. Sec. 2.3.5)

$$\begin{aligned} \mathbf{T} &= \mathcal{DSHT}\{\text{diag}\{g(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\})\} \mathbf{Y}_N(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\})\} \\ &= \mathbf{Y}_N^\dagger(\boldsymbol{\Theta}) \text{diag}\{g(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\})\} \mathbf{Y}_N(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\}). \end{aligned} \quad (3.8)$$

The discrete version of the gain function $g(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\})$ is written as

$$\mathbf{g}(\mathcal{T}^{-1}\{\boldsymbol{\Theta}\}) = [g(\mathcal{T}^{-1}\{\boldsymbol{\theta}_1\}), \dots, g(\mathcal{T}^{-1}\{\boldsymbol{\theta}_L\})]^T. \quad (3.9)$$

3.1.1 Transformation using t -designs

To achieve a low computational effort, a small number of sampling points would be beneficial. However, the number of sampling points must at least be $L \geq (N + 1)^2$. Moreover, the condition number of $\mathbf{Y}(\boldsymbol{\Theta}_N)$ needs to be sufficiently small to avoid numerical errors. The most pragmatic choice of sampling was presented by Hardin and Sloane [HS96], who provide coordinates $\boldsymbol{\Theta}_t$ for various spherical t -designs. For a transform of the Ambisonic order N , we would need a t -design of $t \geq 2N$. The 21-design available from [HS96] with $L = 240$ points allows the use up to Ambisonic order $N = 10$.

The usage of t -designs provides \mathcal{DSHT} without any pseudo-inversion but simple transposition (cf. Eq. (2.21))

$$\mathbf{T} = \text{diag}\left\{\frac{4\pi}{L}\right\} \mathbf{Y}_N^T(\boldsymbol{\Theta}_t) \text{diag}\{g(\mathcal{T}^{-1}\{\boldsymbol{\Theta}_t\})\} \mathbf{Y}_N(\mathcal{T}^{-1}\{\boldsymbol{\Theta}_t\}). \quad (3.10)$$

Due to its practical advantage, all subsequent transformations use the above Eq. (3.10).

Increased Re-Expansion Order \tilde{N}

Note that in general angular modifications other than rotations and non-neutral gain functions increase the required Ambisonic order. In order to avoid loss of signal energy, we use the increased Ambisonic order \tilde{N} for $\mathbf{Y}_{\tilde{N}}^T(\boldsymbol{\Theta}_t)$ to obtain a thin rectangular transformation matrix

$$\mathbf{T}_{N \rightarrow \tilde{N}} = \text{diag}\left\{\frac{4\pi}{L}\right\} \mathbf{Y}_{\tilde{N}}^T(\boldsymbol{\Theta}_t) \text{diag}\{\mathbf{g}(\mathcal{T}^{-1}\{\boldsymbol{\Theta}_t\})\} \mathbf{Y}_N(\mathcal{T}^{-1}\{\boldsymbol{\Theta}_t\}). \quad (3.11)$$

The required re-expansion order \tilde{N} will be discussed later for the individual transformations.

Applying Transformations in Practice

The transformation matrix \mathbf{T} stays constant as long as the angular transformation $\mathcal{T}\{\boldsymbol{\Theta}\}$ and the weighting function $\mathbf{g}(\boldsymbol{\Theta})$ are not changing. Time-varying parameters such as continuous rotation might be necessary in applications. Therefore a transformation matrix has to be computed for each sample. For performance reasons, real-time audio applications usually perform similar operations on blocks of samples. The transformation matrix may be computed for each new sample block and sample-wise interpolation used between consecutive blocks (cf. Fig. 3.2). This approximation seems reasonable for typical block sizes in audio applications ranging from 64-2048 samples.

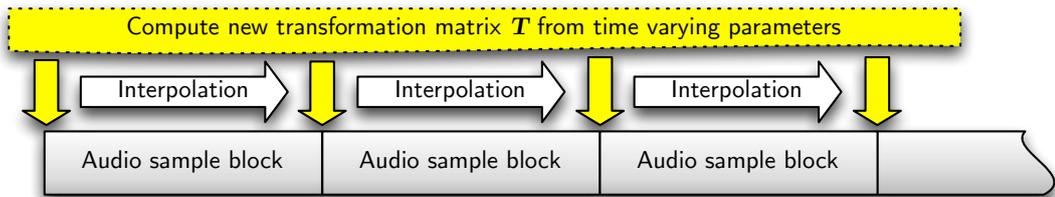


Figure 3.2: Computing a transformation matrix \mathbf{T} for each block of audio samples and interpolating within consecutive \mathbf{T} s.

3.2 Rotation in Three Dimensions

The rotation of Ambisonic signals around the z-axis is fairly easy to implement and can be done using a spherical harmonic rotation matrix \mathbf{T}_r^z following the scheme [Zm02, Son03, Zot09b]

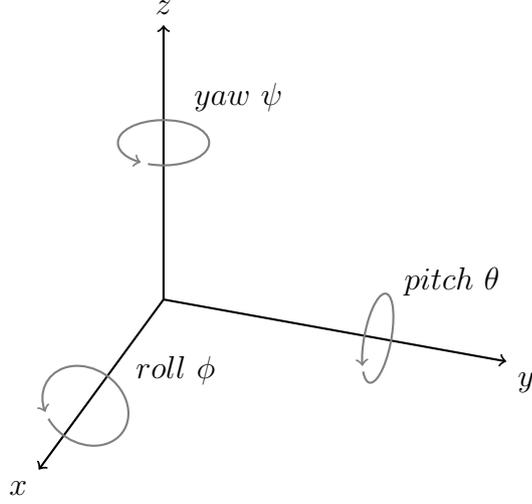


Figure 3.3: Rotation around x, y and z-axis.

$$\mathbf{T}_r^z(\psi) = \left(\begin{array}{c|ccc|cccc|c}
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\
 0 & \cos \psi & 0 & \sin \psi & 0 & 0 & 0 & 0 & 0 & \\
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\
 0 & -\sin \psi & 0 & \cos \psi & 0 & 0 & 0 & 0 & 0 & \\
 \hline
 0 & 0 & 0 & 0 & \cos 2\psi & 0 & 0 & 0 & \sin 2\psi & \\
 0 & 0 & 0 & 0 & 0 & \cos \psi & 0 & \sin \psi & 0 & \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \mathbf{0} \\
 0 & 0 & 0 & 0 & 0 & -\sin \psi & 0 & \cos \psi & 0 & \\
 0 & 0 & 0 & 0 & -\sin 2\psi & 0 & 0 & 0 & \cos 2\psi & \\
 \hline
 \mathbf{0} & & \mathbf{0} & & & & \mathbf{0} & & & \cos 3\psi \\
 & & & & & & & & & \dots
 \end{array} \right). \quad (3.12)$$

The derivation of matrices for a rotation around the x- and y-axis is mathematically more demanding. Zotter proposed a combination of fixed 90° rotations around the y-axis together with variable rotations around the z-axis. In his thesis, Zotter derived a recurrence relation for obtaining the $\mathbf{T}_r^y(90^\circ)$ matrix. Together with Eq. (3.12), this gives us the straight-forward implementation [Zot09b]

$$\mathbf{T}_r^{xyz}(\phi, \theta, \psi) = \mathbf{T}_r^z(\phi + 45^\circ) \mathbf{T}_r^y(90^\circ) \mathbf{T}_r^z(\theta + 180^\circ) \mathbf{T}_r^y(90^\circ) \mathbf{T}_r^z(\psi + 45^\circ) \quad (3.13)$$

For consistency and easy implementation, we want to use the derivations from Sec. 3.1 to obtain $\mathbf{T}_r^{xyz}(\phi, \theta, \psi)$ and describe the rotation in the angular domain.

The rotated unit Cartesian direction vector $\tilde{\boldsymbol{\theta}}$ can be obtained with

$$\tilde{\boldsymbol{\theta}} = \mathcal{T}\{\boldsymbol{\theta}\} = \mathbf{R}(\phi, \theta, \psi) \boldsymbol{\theta}, \quad (3.14)$$

with the rotation matrix in \mathbb{R}^3 [Wei14b]

$$\mathbf{R}(\phi, \theta, \psi) = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{pmatrix}}_{\text{x-axis-rotation(roll)}} \cdot \underbrace{\begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix}}_{\text{y-axis-rotation(pitch)}} \cdot \underbrace{\begin{pmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{z-axis-rotation(yaw)}}. \quad (3.15)$$

We use Eq. (3.10) with neutral directional weighting $g(\boldsymbol{\theta}) = 1$, the inverse angular transformation $\mathcal{T}^{-1}\{\cdot\} = \mathbf{R}^T(\phi, \theta, \psi) \cdot \{\cdot\}$ and obtain our spherical harmonics rotation matrix as

$$\mathbf{T}_r^{xyz} = \text{diag}\left\{\frac{4\pi}{L}\right\} \mathbf{Y}_N^T(\boldsymbol{\Theta}_t) \mathbf{Y}_N(\mathbf{R}^T(\phi, \theta, \psi) \boldsymbol{\Theta}_t). \quad (3.16)$$

Rotation does not increase the order N . Fig. 3.4 shows the sparsity of different rotation matrices.

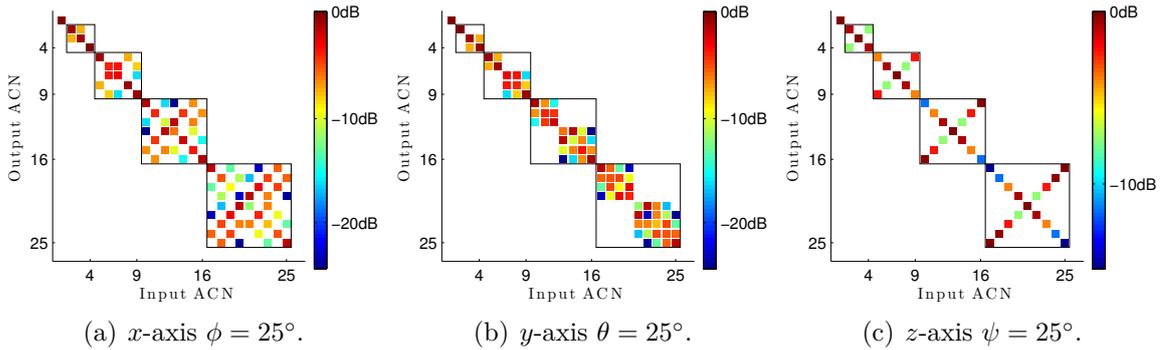


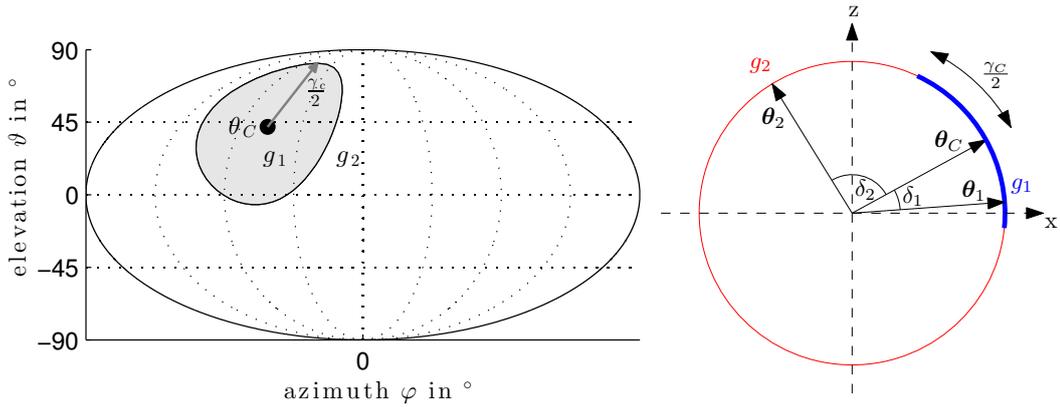
Figure 3.4: Sparsity of $N = 4$ rotation matrices for rotations around different axes.

3.3 Directional Loudness Modifications

Modifying the loudness of specific directions is especially useful for post production of microphone array recording or to adjust for specific playback situation.

We apply the general transformation approach from Sec. 3.1, use Eq. (3.11) with neutral angular mapping $\mathcal{T}\{\boldsymbol{\theta}\} = \boldsymbol{\theta}$, and search for a direction dependent gain function $g(\boldsymbol{\theta})$ to amplify or attenuate certain regions of the surround image.

3.3.1 Spherical Caps



(a) Mollweide projection of the spherical cap.

(b) Two sample directions $\boldsymbol{\theta}_1$ with $\delta_1 < \frac{\gamma_C}{2}$ therefore within, and $\boldsymbol{\theta}_2$ with $\delta_2 > \frac{\gamma_C}{2}$ therefore outside the spherical cap getting scaled with g_1 and g_2 .

Figure 3.5: Spherical cap with center $\boldsymbol{\theta}_C$, size $\frac{\gamma_C}{2}$, gain g_1 inside, and g_2 outside the cap.

We define a spherical cap function of size $\frac{\gamma_C}{2}$ around the center point $\boldsymbol{\theta}_C$ (cf. Fig. 3.5). We use the scalar product of two Cartesian unit vectors to get the angle δ_l between the cap center $\boldsymbol{\theta}_C$ and one t -design sampling point $\boldsymbol{\theta}_l$,

$$\cos \delta_l = \boldsymbol{\theta}_C^T \boldsymbol{\theta}_l, \quad (3.17)$$

and define the gain function using Eq. (3.17) as

$$g(\boldsymbol{\theta}) = u(\boldsymbol{\theta}_C^T \boldsymbol{\theta} - \cos \frac{\gamma_C}{2}), \quad (3.18)$$

with $u(\cdot)$ being the unit step function. Instead of cropping the spherical cap it might be practical to define gains for both regions within the spherical cap and outside with g_1 and g_2 (cf. Fig. 3.5), respectively,

$$g(\boldsymbol{\theta}) = g_1 u(\boldsymbol{\theta}_c^T \boldsymbol{\theta} - \cos \frac{\gamma_c}{2}) + g_2 u(\cos \frac{\gamma_c}{2} - \boldsymbol{\theta}_c^T \boldsymbol{\theta}). \quad (3.19)$$

The sharp edges of this spatial window may introduce high re-expansion orders \tilde{N} depending on the position in the spherical spectrum of those edges. The result of truncating \tilde{N} can be seen in Fig. 3.6 and Fig. 3.7. Other more rounded functions would help to keep the order small but will not be discussed here. Instead a different approach using spherical Slepian functions will be presented in the following section.

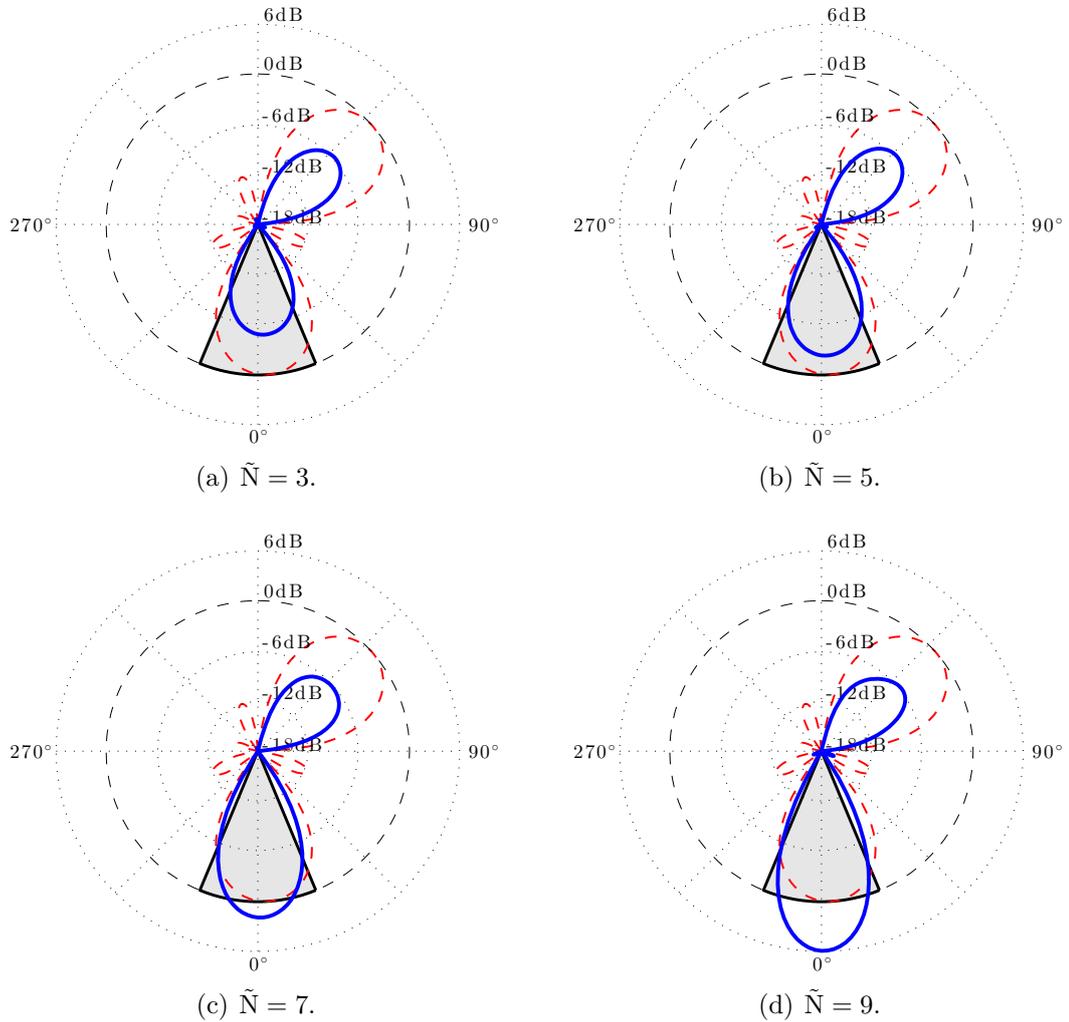


Figure 3.6: Directional loudness modification with spherical cap at $(0^\circ, 0^\circ)$, size $\gamma = 40^\circ$, $g_1 = 6\text{dB}$, $g_2 = -6\text{dB}$, and different re-expansion orders \tilde{N} for 2 virtual sources with $N = 3$.

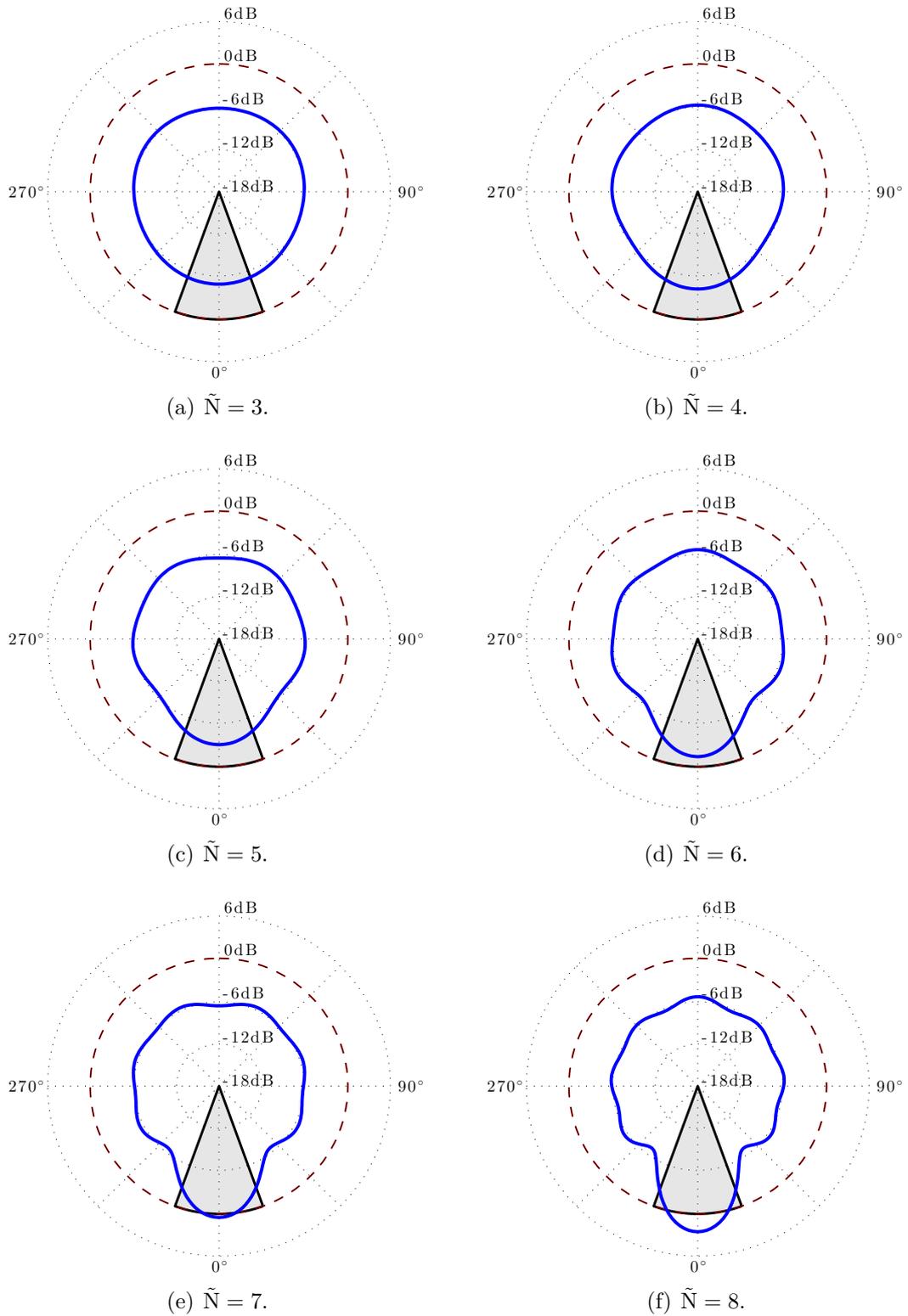


Figure 3.7: Directional loudness modification with spherical cap at $(0^\circ, 0^\circ)$, size $\gamma = 40^\circ$, $g_1 = 6\text{dB}$, $g_2 = -6\text{dB}$, and different re-expansion orders \tilde{N} for omni-directional signal.

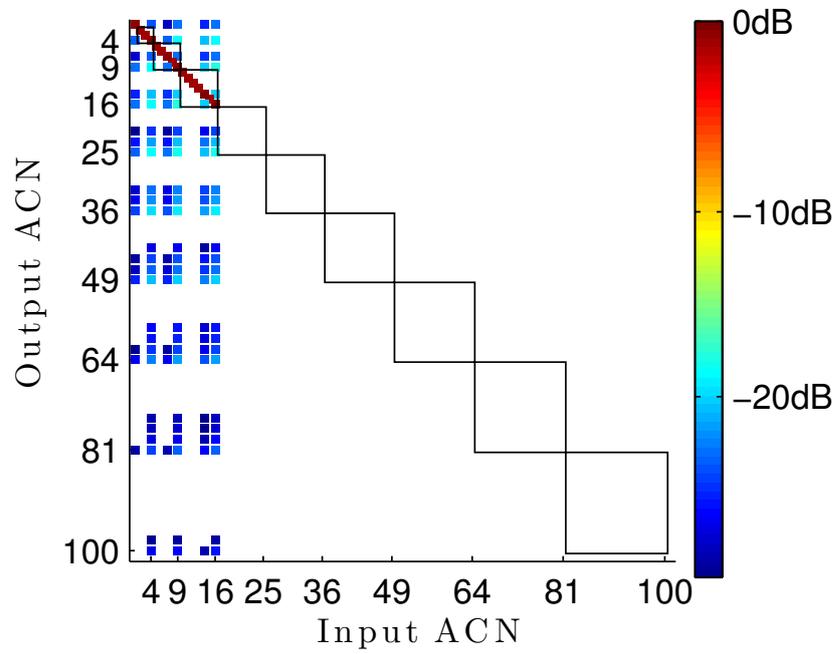


Figure 3.8: Sparsity of the truncated transformation matrix for directional loudness modification using a spherical cap at $(0^\circ, 0^\circ)$, size $\gamma = 40^\circ$, $g_1 = 6\text{dB}$, $g_2 = -6\text{dB}$.

3.3.2 Spherical Slepian Functions

Suppressing signal components from certain directions can also be achieved by using spherical Slepian functions. Spherical Slepian functions are linear combinations of spherical harmonics and have so far been used for the computation of Ambisonic decoders for hemispherical playback layouts [ZPN12], obtaining a reduced set of Ambisonics transmission channels [PZ09], or for the interpolation of spherical measurement data [ZP12]. Sun et al. [SS11] use a set of modified basis functions obtained by convex optimization to suppress signals from specified directions. Obtaining such modified basis functions is addressed in this section.

A well-explored usage of spherical Slepian functions comes from the field of geodesy. Satellite measurements from the earth often lack data from the pole regions [SF06]. Using spherical harmonics within limited regions on the sphere yields to ill-posed inversion problems and therefore spherical Slepian functions are derived which form an orthogonal set of basis functions within limited portions on the unit sphere.

We want to crop out a part of the surround signal by choosing a subset of the sphere $S^2 \subset \mathbb{S}^2$

$$\tilde{\phi}_{\tilde{N}}(t) = \int_{S^2} \mathbf{y}_{\tilde{N}}(\boldsymbol{\theta}) \mathbf{y}_{\tilde{N}}^T(\boldsymbol{\theta}) d\boldsymbol{\theta} \phi_N(t). \quad (3.20)$$

In contrast to the integral over the full spherical surface \mathbb{S}^2 in Eq. (2.11), the integral

$$\int_{S^2} \mathbf{y}_{\tilde{N}}(\boldsymbol{\theta}) \mathbf{y}_{\tilde{N}}^T(\boldsymbol{\theta}) d\boldsymbol{\theta} = \mathbf{G} \quad (3.21)$$

is not the identity matrix and we obtain \mathbf{G} by using the cap function from Eq. (3.18) and $DSHT$ (cf. Eq. (3.10))

$$\begin{aligned} \mathbf{G} &= \int_{\mathbb{S}^2} \mathbf{y}_{\tilde{N}}(\boldsymbol{\theta}) \text{diag}\{\mathbf{g}(\boldsymbol{\theta})\} \mathbf{y}_{\tilde{N}}^T(\boldsymbol{\theta}) d\boldsymbol{\theta}, \\ &= \text{diag}\left\{\frac{4\pi}{L}\right\} \mathbf{Y}_{\tilde{N}}^T(\boldsymbol{\Theta}_t) \text{diag}\{\mathbf{g}(\boldsymbol{\Theta}_t)\} \mathbf{Y}_{\tilde{N}}(\boldsymbol{\Theta}_t). \end{aligned} \quad (3.22)$$

Singular value decomposition on \mathbf{G} yields [ZPN12]

$$\mathbf{G} = \mathbf{U} \text{diag}\{[\sigma_i]_{1\dots(N+1)^2}\} \mathbf{V}^T, \quad (3.23)$$

where \mathbf{U} and \mathbf{V}^T contain the left and right singular vectors and σ_i are the corresponding eigenvalues. The eigenvalues indicate how much energy is concentrated in the corresponding basis functions.

We obtain our transformation matrix by selecting a subset of the eigenvectors

$$\mathbf{T} = \mathbf{U} \text{diag}\{\{\varsigma_i\}_{1\dots(N+1)^2}\} \mathbf{V}^T, \quad (3.24)$$

where ς is scaling the eigenvectors. In this case we scale all eigenvectors corresponding to eigenvalues above a fraction α of the largest eigenvalue $\sigma_i > \alpha \sigma_1$ by g_1 and all eigenvectors with $\sigma_i < \alpha \sigma_1$ by g_2 , $0 < \alpha < 1$

$$\varsigma_i = g_1 u(\sigma_i - \alpha \sigma_1) + g_2 u(\alpha \sigma_1 - \sigma_i). \quad (3.25)$$

Fig. 3.9 shows the sparsity and polar plots for directional loudness modification using spherical Slepian functions. This transformation allows to freely choose the re-expansion order $\tilde{N} \geq N$ without having to truncate signal energy from higher orders.

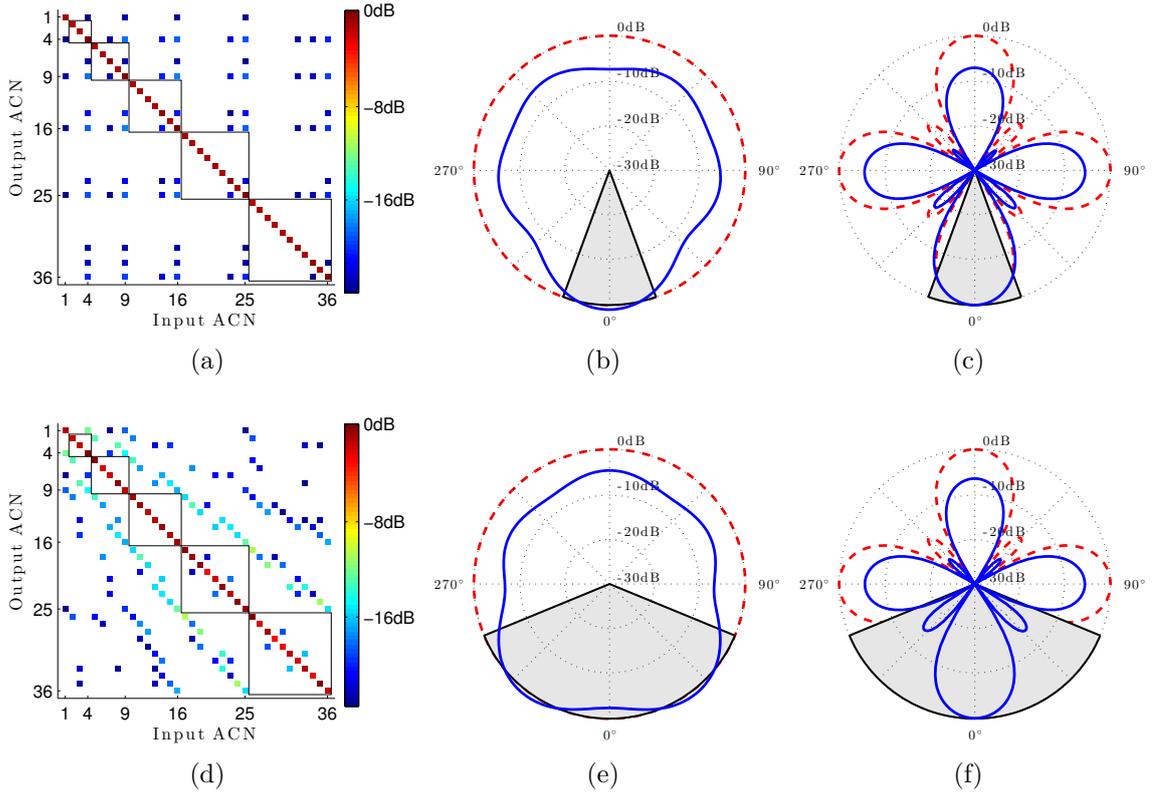


Figure 3.9: Directional loudness modification using spherical Slepian functions with $N = \tilde{N} = 5$, $\alpha = \frac{1}{2}$, $g_1 = 0\text{dB}$ and $g_2 = -6\text{dB}$, (a)-(c) spherical cap at $(0^\circ, 0^\circ)$ and size $\gamma = 40^\circ$, (d)-(f) spherical cap at $(0^\circ, 0^\circ)$ and size $\gamma = 135^\circ$, (a) and (d) show the sparsity of the transformation matrix, Ambisonic order does not increase, (b) and (e) show polar diagrams for omni directional signal, (c) and (f) show polar diagrams for 4 virtual sources at cardinal directions, dashed lines indicate unmodified signals.

3.4 Warping

Warping describes an angular transformation for extending and narrowing the surround image. This effect is useful in order to adjust the width of a recorded sound scene or to compensate for playback situations with reduced coverage of the sphere. Using a hemispherical loudspeaker array to playback recordings with sound sources from below the equator is obviously not possible without loss of signal content. Warping can be used to pull the surround image from the lower hemisphere towards the zenith and therefore move sound sources into the area covered by loudspeakers.

Mid-Side Recording

The well known mid-side (MS) stereo recording technique allows to adjust the stereo spread of the recording, and it is very similar to the warping operation described here. MS recording uses one omnidirectional or cardioid mid microphone and one coincident figure-of-eight microphone facing to the left (cf. Fig. 3.10). By linear combination of the two microphone signals the left and right channel of the stereo signal is acquired

$$L = M + \lambda S, \quad (3.26)$$

$$R = M - \lambda S.$$

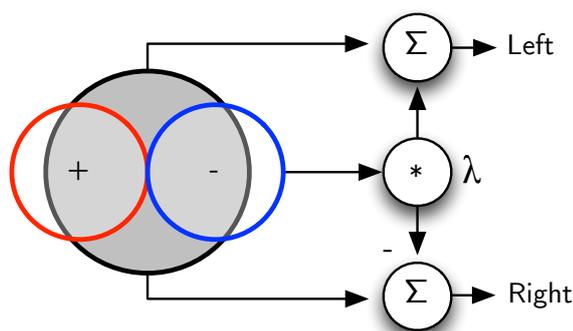


Figure 3.10: Mid-side recording using one coincidental omni directional and figure-of-eight microphone allows to adjust the recording angle by varying the parameter λ .

Gerzons' Forward Dominance

Gerzon [GB92] presented the azimuthal warping operation (forward dominance) for first-order Ambisonics in order to adjust the frontal-back balance

$$\begin{aligned} W' &= \frac{1}{2}\left(\lambda + \frac{1}{\lambda}\right)W + \frac{1}{\sqrt{8}}\left(\lambda - \frac{1}{\lambda}\right)X, \\ X' &= \frac{1}{2}\left(\lambda + \frac{1}{\lambda}\right)X + \frac{1}{\sqrt{2}}\left(\lambda - \frac{1}{\lambda}\right)W, \\ Y' &= Y, \end{aligned} \tag{3.27}$$

whereby λ is the dominance parameter with $2 > \lambda > 1$ for forward dominance, and $1 > \lambda > 0$ backward dominance.

This would translate to the forward dominance transformation matrix T_{fwd} for first-order Ambisonics using the convention in Sec. 2.3.1

$$T_{\text{fwd}} = \begin{pmatrix} \frac{1}{2}\left(\lambda + \frac{1}{\lambda}\right) & 0 & 0 & \frac{\sqrt{3}}{4}\left(\lambda - \frac{1}{\lambda}\right) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{\sqrt{3}}\left(\lambda - \frac{1}{\lambda}\right) & 0 & 0 & \frac{1}{2}\left(\lambda + \frac{1}{\lambda}\right) \end{pmatrix} \tag{3.28}$$

Warping for Higher-Order Ambisonics

Performing the warping operation in the angular domain was already proposed by Malham:

"One further possibility which would bear investigation would be to spatially oversample the soundfield using a sufficient number of sampling points to avoid spatial aliasing and then to produce a new soundfield by resampling the points using a suitable warping function." [Mal03]

Zotter and Pomberger [ZP11, PZ11] described warping for higher-order Ambisonics and derived recurrence relations for computing the transformation matrix. They mention the necessity for a magnitude emphasis as correction for the enlargement of sources after applying warping. Emphasis is caused due to the playback of stretched regions through a larger number of loudspeakers, compressed regions will be played back by fewer loudspeakers and appear quieter without compensation. Therefore

non of our modifiers $g(\boldsymbol{\theta})$ and $\mathcal{T}\{\boldsymbol{\theta}\}$ will be neutral in Eq. (3.11). Note that we are searching for invertible modifiers in order to make use of the practical advantages from Eq. (3.11).

Due to simplicity we will describe warping only for the elevation angle, but the operation can be performed along any direction by pre- and post rotation.

The elevation angle ϑ is obtained from our Cartesian unit vector using Eq. (2.2). To simplify the derivation we use the substitution [PZ11],

$$\begin{aligned} \mu &= \sin(\vartheta), & \text{original,} & \\ \tilde{\mu} &= \sin(\tilde{\vartheta}), & \text{warped,} & \end{aligned} \tag{3.29}$$

and restrict ourselves to monotonically rising warping curves $\frac{\partial \tilde{\mu}}{\partial \mu} \geq 0$ that map μ of the interval $[-1, 1]$ to $\tilde{\mu}$ covering the interval $[-1, 1]$.

3.4.1 Warping Towards a Pole

As proposed by Gerzon [GB92] and in [ZP11], a bilinear transform provides a useful warping transformation between μ and $\tilde{\mu}$

$$\tilde{\mu} = \mathcal{T}\{\mu\} = \frac{\mu + \alpha}{1 + \alpha\mu}. \tag{3.30}$$

The operation is neutral for $\alpha = 0$, and depending on the sign of α , it elevates or lowers the equator $\vartheta_0 = 0$ of the original surround image to $\tilde{\vartheta}_0 = \arcsin \alpha$ for any α between $-1 < \alpha < 1$ (cf. Fig. 3.11(a)).

Loudness Compensation

To preserve the loudness of sounds within the stretched and squeezed parts of the surround sound, a gain correction has to be applied before (pre-emphasis) or after (de-emphasis) angular warping.

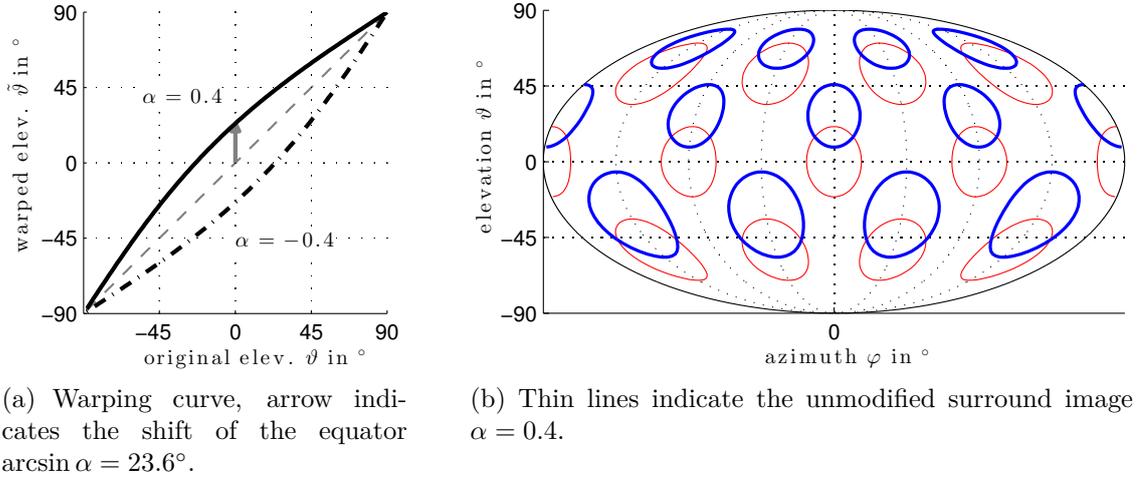


Figure 3.11: Schemes for warping towards a pole.

We describe the enlargement σ of a virtual source by the derivative of the angular transformation [PZ11],

$$\begin{aligned} \sigma &= \frac{\partial \tilde{\mu}}{\partial \mu} = \frac{1}{\frac{\partial \mu}{\partial \tilde{\mu}}}, \\ &= \frac{1 - \alpha^2}{(1 + \alpha\mu)^2} = \frac{(1 - \alpha\tilde{\mu})^2}{1 - \alpha^2}, \end{aligned} \quad (3.31)$$

and achieve energy-preserving warping by formulating the de-emphasis after warping as

$$g(\tilde{\mu}) = \frac{1}{\sqrt{\sigma}} = \frac{\sqrt{1 - \alpha^2}}{1 - \alpha\tilde{\mu}}. \quad (3.32)$$

It can be seen from Fig. 3.12 that the matrix distributes a part of the signal energy to higher orders. The required re-expansion order \tilde{N} is exemplary shown in Tab. 3.1. Applying this transformation matrix to an Ambisonically encoded signal yields the polar plots in Fig. 3.13.

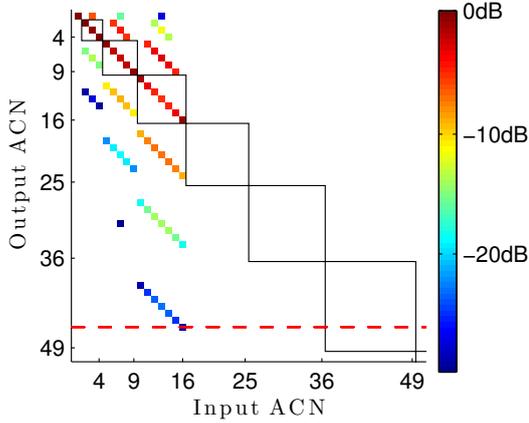
(a) Without loudness compensation.

N	5°	10°	15°	20°	25°	30°
1	2	2	2	3	3	3
2	3	3	4	4	5	6
3	4	5	5	6	7	8
4	5	6	7	8	9	10
5	6	7	8	9	11	12

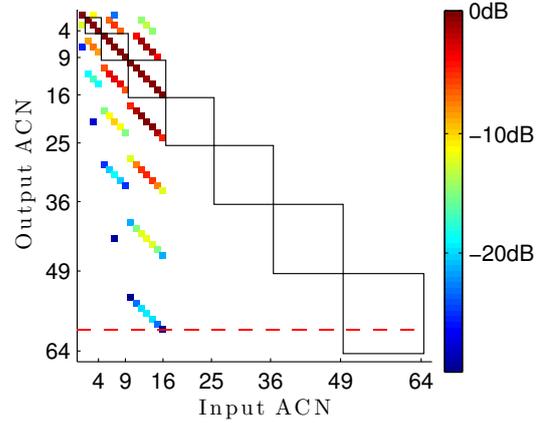
(b) With loudness compensation.

N	5°	10°	15°	20°	25°	30°
1	2	2	3	3	4	5
2	3	4	4	5	6	7
3	4	5	6	7	8	9
4	5	6	7	9	10	12
5	7	8	9	10	12	14

Table 3.1: Warping towards a pole, required re-expansion order \tilde{N} for different input signal orders N and warping angles $\tilde{\vartheta}_0 = \arcsin(\alpha)$. (Order above which re-expansion uses coefficients less than -30dB.)



(a) Without loudness compensation.



(b) With loudness compensation.

Figure 3.12: Warping towards the north pole, sparsity of \mathbf{T} . Dashed line indicates the last entry above -30dB. For $\alpha = 0.4$ and $N = 3$ the re-expansion order $\tilde{N} = 7$ has to be used in order to avoid errors > -30 dB.

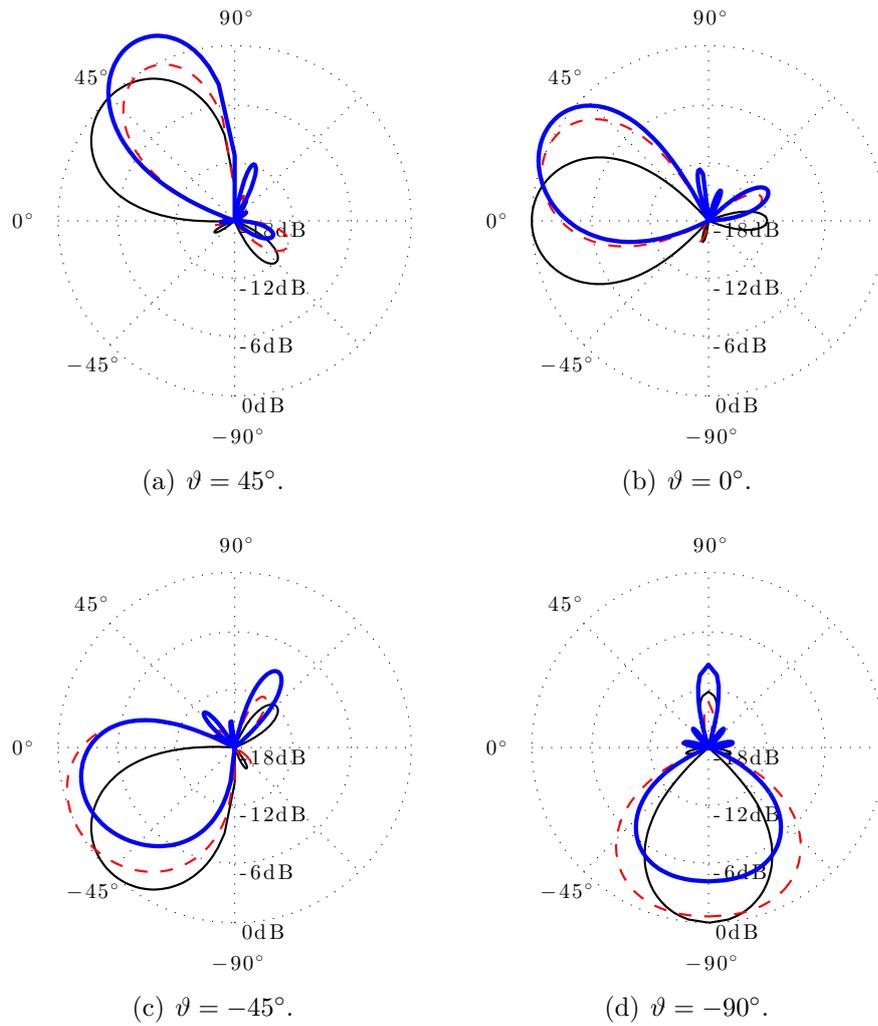


Figure 3.13: Warping towards the north pole, polar plots for a virtual source at different elevations ϑ , encoding order $N = 3$, $\tilde{N} = 7$, $\alpha = 0.4$, dashed line indicates warping without loudness compensation.

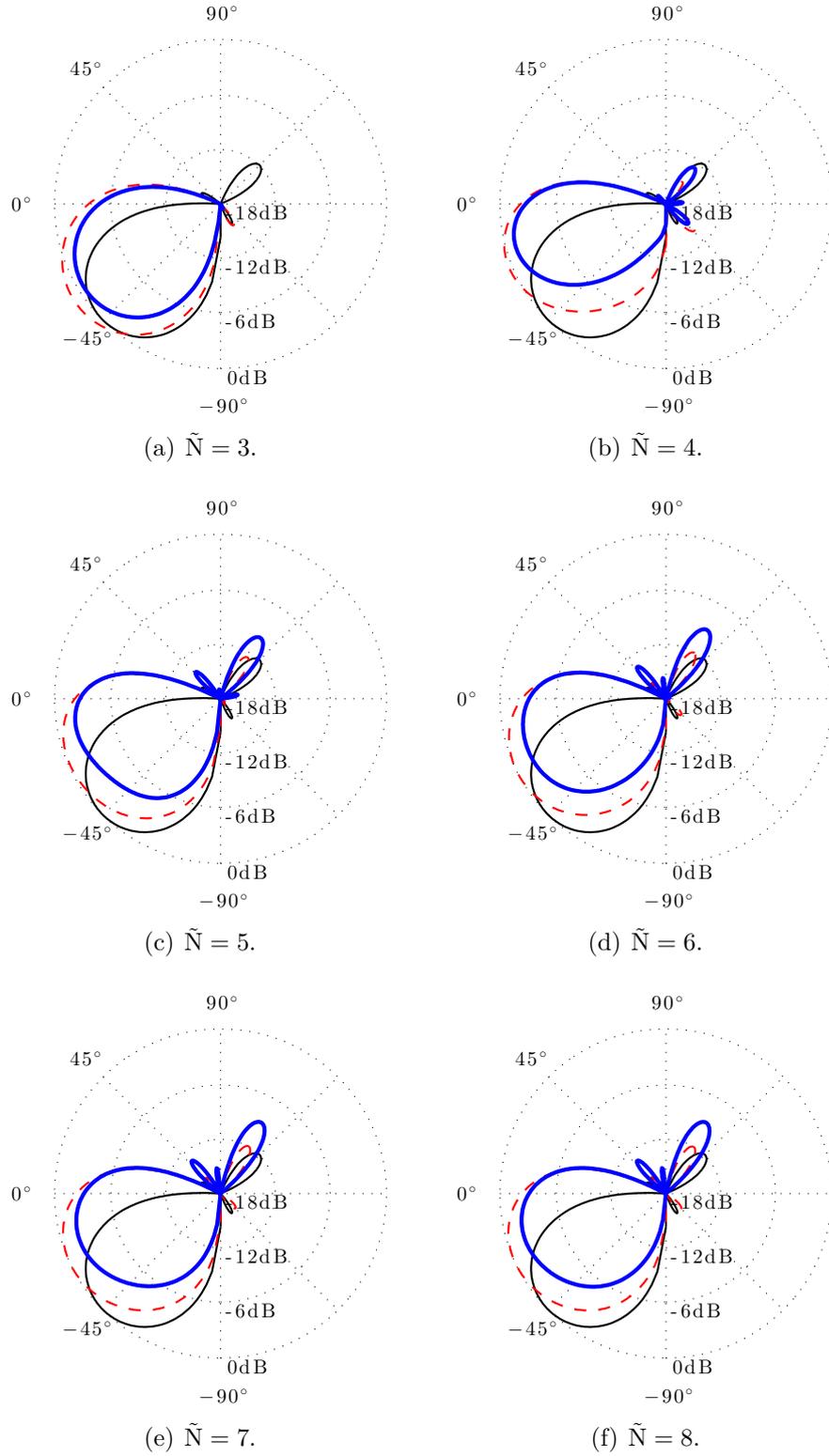
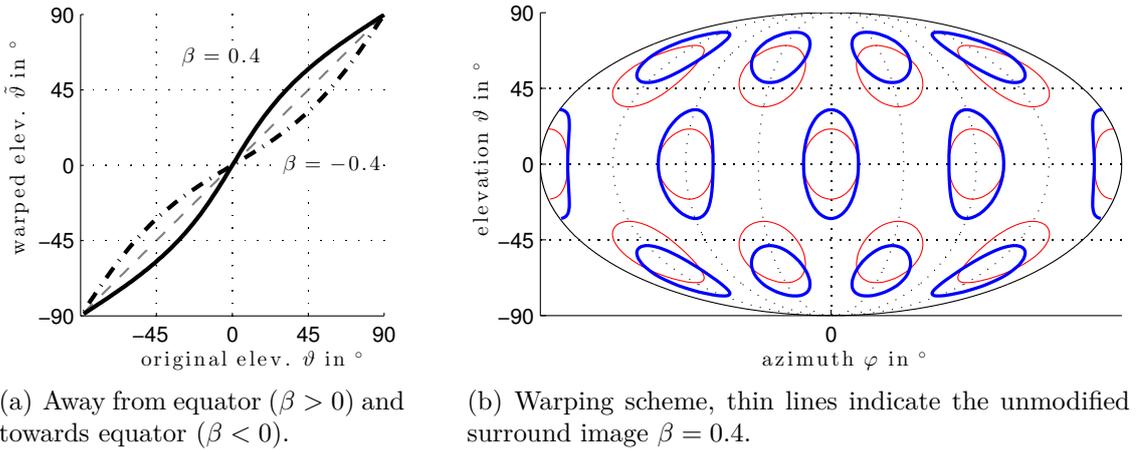


Figure 3.14: Impact of truncating the re-expansion order \tilde{N} for warping towards the north pole, polar plots for a virtual source at $\vartheta = -45^\circ$, $\alpha = 0.4$, encoding order $N = 3$ and different \tilde{N} .

3.4.2 Warping Towards and Away from the Equator

It is worth looking into an alternative warping curve that does not shift the equator, but moves sources towards the equator or away from it. This can be useful in order to adapt full periphonic recordings to circular loudspeaker arrangements. The following warping equation pushes surround content away from the equator towards the poles for $\beta > 0$, or pulls the surround image towards the equator for $\beta < 0$ (cf. Fig. 3.15),

$$\tilde{\mu} = \mathcal{T}\{\mu\} = \begin{cases} \frac{(|\beta|-1) + \sqrt{(|\beta|-1)^2 + 4|\beta|\mu^2}}{2|\beta|\mu}, & \text{for } \beta > 0, \\ \frac{(1-|\beta|)\mu}{1-|\beta|\mu^2}, & \text{for } \beta < 0. \end{cases} \quad (3.33)$$



(a) Away from equator ($\beta > 0$) and towards equator ($\beta < 0$).

(b) Warping scheme, thin lines indicate the unmodified surround image $\beta = 0.4$.

Figure 3.15: Warping towards and away from the equator.

Inspecting Eq. (3.31) and Eq. (3.32), we can similarly derive the de-emphasis for this warping operation as

$$g(\mu) = \left(\frac{1 - |\beta|\mu^2}{\sqrt{(1 - |\beta|)(1 + |\beta|\mu^2)}} \right)^{\text{sgn}\{\beta\}}. \quad (3.34)$$

The exponent denotes that, for negative β , de-emphasis uses the reciprocal value of the expression in brackets.

The required re-expansion order \tilde{N} can exemplarily found in Tab. 3.2. Fig. 3.16 shows the polar diagram for different truncation of \tilde{N} .

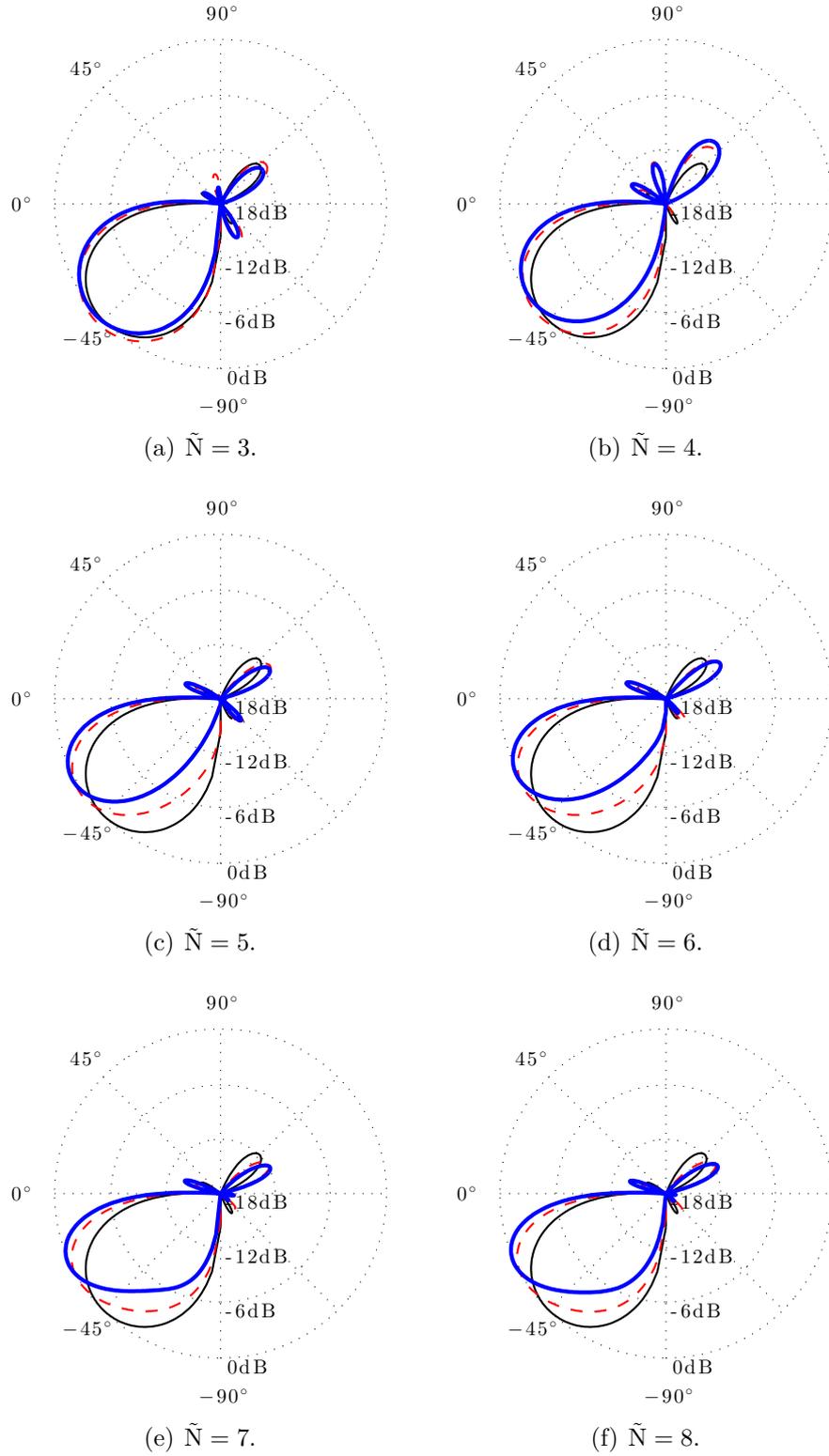


Figure 3.16: Impact of truncating the re-expansion order \tilde{N} for warping towards the equator, polar plots for a virtual source at $\vartheta = -45^\circ$, $\beta = -0.4$, encoding order $N = 3$ and different \tilde{N} .

(a) Without loudness compensation.							(b) With loudness compensation.						
N	0.087	0.18	0.26	0.36	0.45	0.55	N	0.087	0.18	0.26	0.36	0.45	0.55
1	3	3	3	3	5	5	1	3	3	5	5	7	9
2	4	4	4	6	6	8	2	4	4	6	8	10	12
3	5	7	7	6	7	13	3	5	7	7	9	11	13
4	6	6	8	10	12	14	4	6	8	10	12	14	15
5	7	9	9	11	15	16	5	7	9	11	13	15	16

Table 3.2: Warping towards the equator, required re-expansion order \tilde{N} for different input signal orders N and warping parameters $-\beta$. (Order above which re-expansion uses coefficients less than -30dB.)

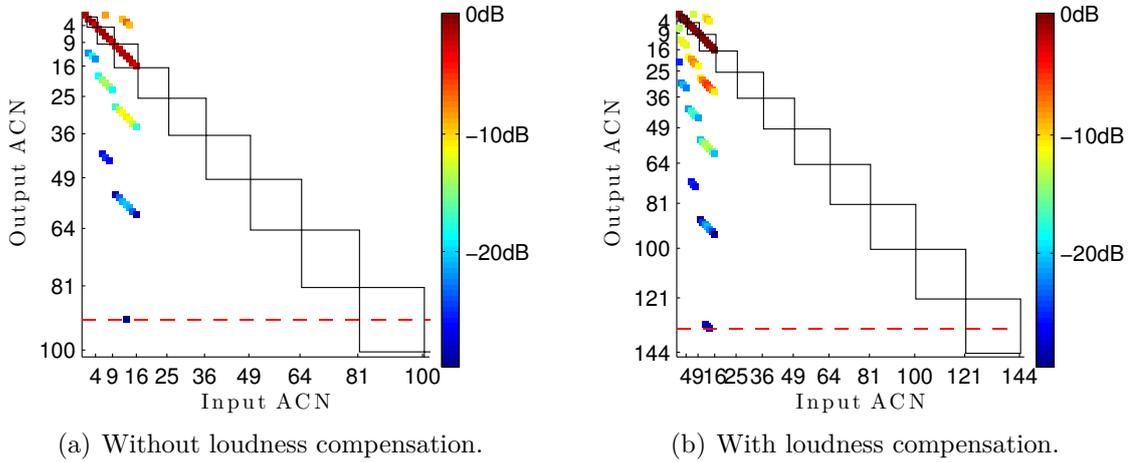


Figure 3.17: Warping towards the equator, sparsity of T . Dashed line indicates the last entry above -30dB. For $\beta = -0.4$ and $N = 3$ the re-expansion order $\tilde{N} = 11$ with loudness compensation has to be used in order to avoid errors > -30 dB.

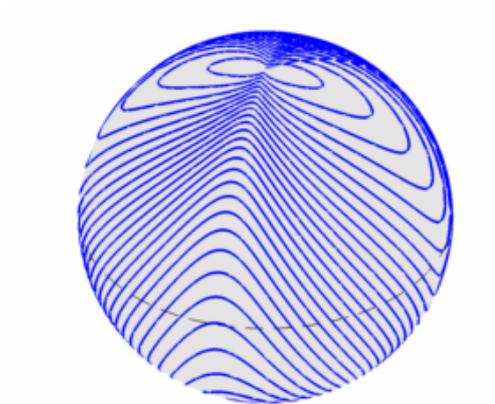
3.5 Other Transformation Curves

The transformation approach from Sec. 3.1 can be used to derive transformation matrices for any directional modification on the spherical surface. For artistic purposes one can think about many possible distortion curves. The Philips Pavilion (cf. Fig. 3.18) constructed from hyperbolic paraboloids by Le Corbusier and Iannis Xenakis served as inspiration for the following transformation using an azimuth dependent warping towards a pole (cf. Fig. 3.19)

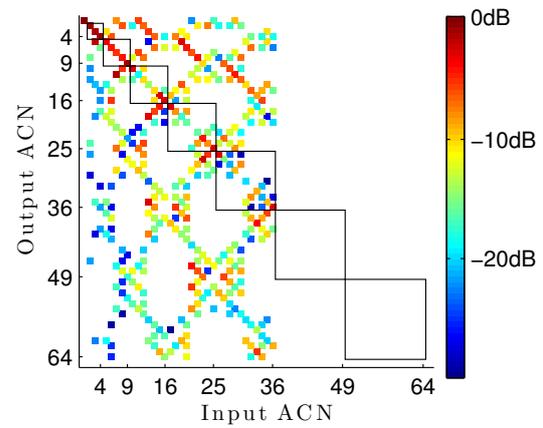
$$\begin{aligned}\mu &= \cos \vartheta, \\ \alpha &= 0.8 \sin 2\phi, \\ \tilde{\mu} &= \frac{\alpha + \mu}{1 + \mu\alpha}, \\ \tilde{\vartheta} &= \arccos \tilde{\mu}.\end{aligned}\tag{3.35}$$



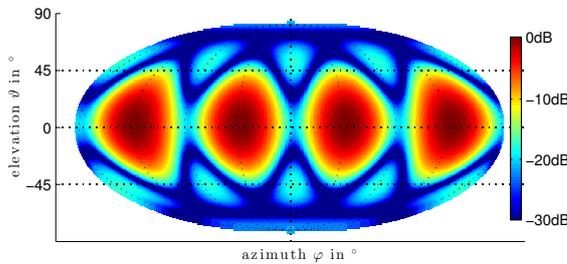
Figure 3.18: Philips Pavilion at the world expo 1958 in Brussels formed from hyperbolic paraboloids, designed by Le Corbusier and Iannis Xenakis for the performance of spatialized electronic music. [© wikimedia commons / Wouter Hagens]



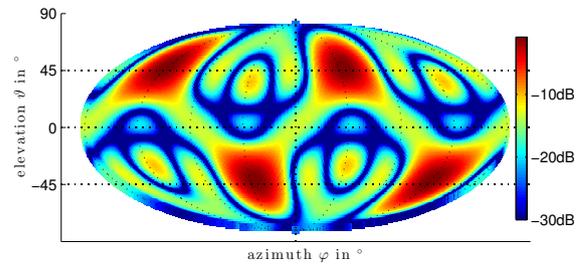
(a) Distortion scheme, lines indicate mapping of constant elevation levels.



(b) Sparsity of \mathbf{T} , truncated at $\tilde{N} = 7$.



(c) Original surround signal.



(d) Modified surround signal.

Figure 3.19: Alternative angular distortion, (c) and (d) show max-re weighted [ZF12] virtual sources from cardinal directions with $N = 5$ and $\tilde{N} = 7$.

3.6 Transformations through Symmetry Properties

Spherical harmonics exhibit several symmetries with regard to the coordinate axes as shown in [Cha09b] and [Zot09b]. We can use those symmetries to efficiently perform mirroring operations on Ambisonic signals.

We will use order n and degree m of the spherical harmonics Y_n^m (cf. Sec. 2.3) to describe their symmetry properties.

In general, sign-inverting signal components with odd symmetry regarding one axis results in mirroring the surround image with respect to this axis. Weighting of odd-symmetric signal components with zero can be used to enforce symmetry (merge signals from opposite directions). This however yields to energy loss which has to be compensated for by amplifying the even-symmetric components.

Mirroring Front-Back

Spherical harmonic components with

$$\begin{aligned} & ((m < 0) \wedge (m \text{ is even})) \\ & \vee ((m \geq 0) \wedge (m \text{ is odd})). \end{aligned} \tag{3.36}$$

exhibit odd symmetry with regard to the x -axis (cf. Fig. 3.20(a)) and therefore may be sign-inverted to achieve a mirroring of the front and back of an Ambisonic signal.

Mirroring Left-Right

Spherical harmonic components with

$$m < 0 \tag{3.37}$$

exhibit odd symmetry (cf. Fig. 3.20(b)) with regard to y and therefore may be sign-inverted to exchange the left- and right side of the surround image.

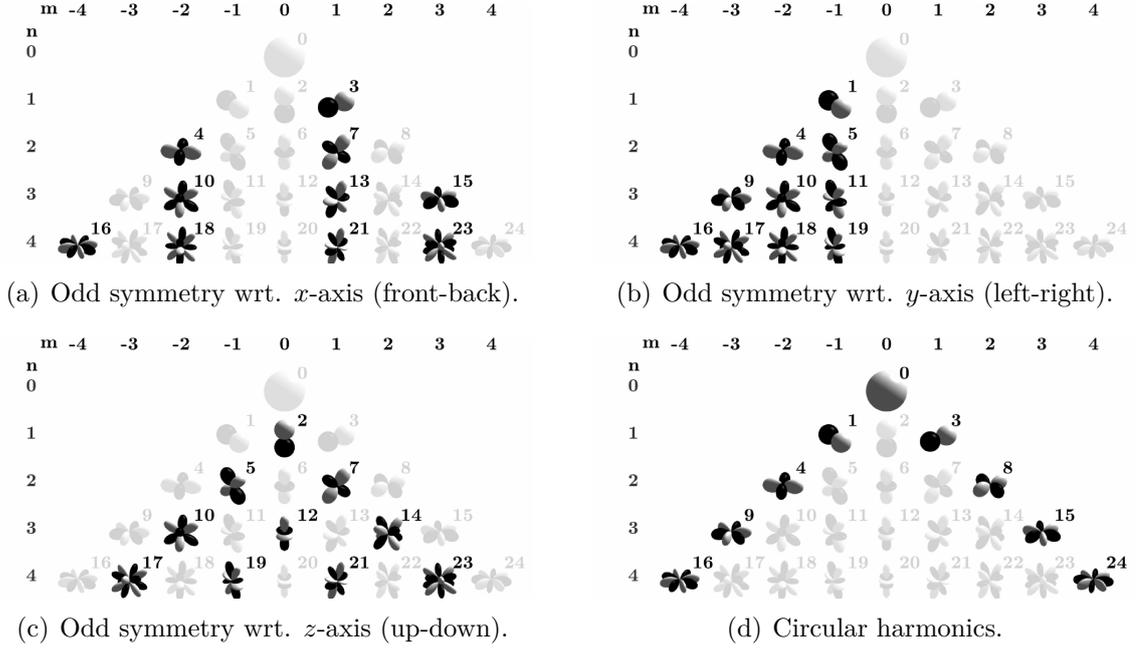


Figure 3.20: Spherical harmonics symmetry properties.

Mirroring Top-Bottom

Exchanging the upper- and lower hemisphere can be done by sign-inverting spherical harmonic components with odd symmetry in z (cf. Fig. 3.20(c)), which is the case whenever

$$(n + m) \text{ is odd.} \quad (3.38)$$

The corresponding transformation matrix for mirroring with regard to z is

$$\mathbf{T}_{\text{mirror}}^z = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{0} \\ \hline \mathbf{0} & \ddots \end{pmatrix}. \quad (3.39)$$

Chapter 4

Practical Contributions

While numerous publications regarding Ambisonics have greatly improved the technological aspects, there is still a lack of tools for creative work in spatial audio using Ambisonic spatialization technique. Rumor even has it that some authors did not even test the practical application of the technology they developed. The responsible goal of this thesis was therefore not just to derive theoretical scenarios, but to implement, test and apply all algorithms.

Most sound artists prefer to produce their music within a digital audio workstation (DAW), which offers great tools for cutting and mixing a multitude of sound objects into the final distribution format. Using Ambisonics in DAWs requires the mixing engine to be capable of handling a large number of channels per track. Usually the maximum number of channels per track is defined by the standard surround formats such as *5.1*. To overcome the limitation of DAWs, most toolkits for spatial audio were developed in graphical programming environments, such as *Pure Data* and *MaxMSP*, which basically have no restriction concerning the number of channels and offer quick development and maintenance cycles. However, not only is the integration of such external spatialization toolboxes into a DAW challenging for the developer, these toolboxes are often also challenging to use for non-experts. Audio channels and control data have to be sent between the applications and complex routing scenarios need to be re-established and carefully archived. This frequently causes difficulties when sharing projects. Moreover it is difficult to render tracks offline, therefore the export often happens in real time and small adjustments might be time consuming.

During the work it became clear that software had to be developed covering the whole signal chain starting from encoding sound sources up to decoding Ambisonic recordings and playing them back through loudspeakers and headphones. While

working with multichannel loudspeaker or microphone arrays in practice, it becomes necessary to perform basic manipulations on the individual audio channels. Most audio effect plug-ins are restricted to process two channels only, some few can be used for standard surround formats such as 5.1. Out of this need, some standard audio processors have been developed without limitation about the number of input and output channels. These multichannel effect plug-ins (*mcfx*) will be described in Sec. 4.2.

The software developed within this thesis (Fig. 4.1) is meant to do away with the above-mentioned shortcomings. It has been developed as an open source project in C++ using the *JUCE* [Sto14] framework which supports building audio plug-ins in all major formats and standalone audio applications for Windows, MacOS, Linux, iOS and Android. Vector and matrix manipulations are performed using *Eigen* [GJ⁺10], a C++ template library for linear algebra. The spherical harmonics are implemented with recursion formulas that can be found in [Zot09b].

Currently the author provides¹ VST and standalone binaries for Windows and MacOS. Using flexible DAWs such as *Reaper* [Coc14] or *Ardour* [Dav14a] allow for a single software controlling the entire production cycle starting from panning the source signals, mixing microphone array recordings until decoding the playback signals and adjusting the loudspeaker signals. The plug-ins can also be used in graphical patching environments such as *MaxMSP* [Cyc14] and *Bidule* [Plo14] or as standalone applications and inter-connected using *Jack* [Dav14b].

4.1 ambiX Audio Plug-ins

The name ambiX is taken from the standardization attempt in [NZDS11] and the software uses the proposed conventions. The maximum Ambisonic order N for all *ambiX* plug-ins can be defined at compile time and is only limited by the maximum number of channels per track of the host software. Currently *Reaper* supports 64 channels per track resulting in the maximum usable Ambisonic order $N = 7$ for full periphony.

¹<http://www.matthiaskronlachner.com>

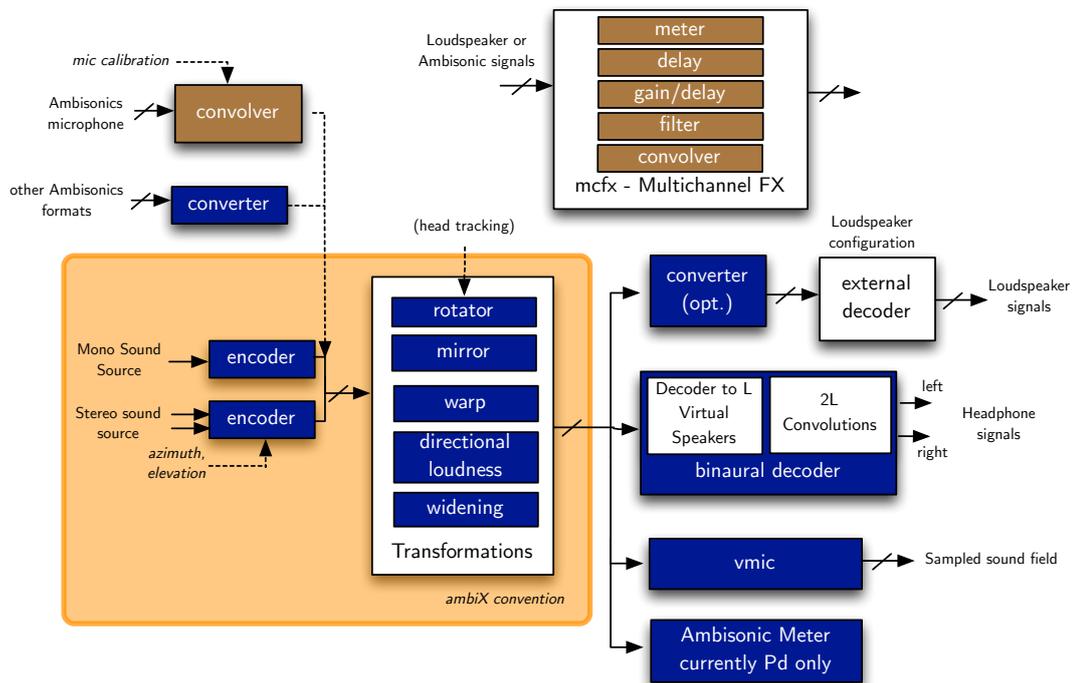
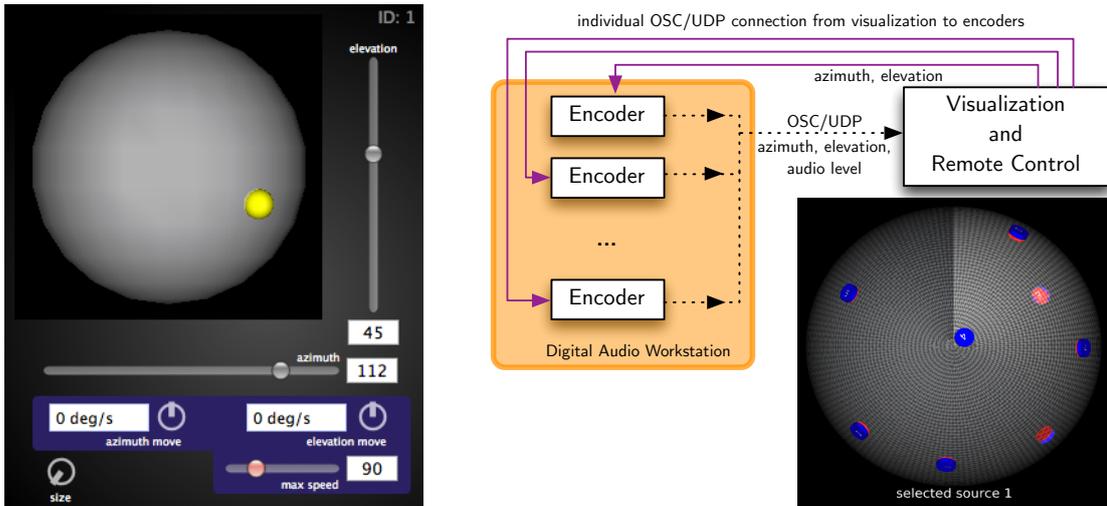


Figure 4.1: ambiX and mcfx plug-ins overview.

Encoding Source Signals

The *ambix_encoder* plug-in is used for panning mono or multichannel sound sources. The position can either be controlled by sliders representing azimuth and elevation or by dragging a yellow ball (cf. Fig. 4.2(a)) on the 3D representation of a sphere. Additionally it is possible to define the speed for horizontal and vertical rotations. This feature allows quick access to basic movements without the need of drawing automation curves. The encoding plug-in has some more automation features that are discussed in [Kro13].

Keeping track of a large number of source positions can be challenging and the user might want to have an alternative to opening the graphical user interface of each encoder individually. To overcome this problem, the *ambix_encoder* plug-in has built in Open Sound Control (OSC) functionality, which allows to control the source position from outside the host application (cf. Fig. 4.2(b)). An automatically launched application monitors and controls the position of several encoders simultaneously. During a mixing session, the user can keep the overview of the spatial sound scene. It is just a matter of taste whether this application may run on the same computer or another device connected via network.



(a) Graphical user interface of the encoder, intuitive panning by clicking on the surface of the sphere, move parameters give quick access to rotations.

(b) Bi-directional OSC communication for monitoring and controlling multiple encoders, this application may run on the same computer or on a different device connected via network..

Figure 4.2: *ambix_encoder* is used for panning a sound source on the surface of the sphere.

Source Widening

The plug-in *ambix_widening* allows to gradually increase the diffuseness of the Ambisonic surround image. The algorithm applies a frequency-dependent rotation through a sparse FIR filter, yielding a frequency-dispersed direction of arrival (Fig. 4.3) [ZFKC14]. By removing the acausal part of the impulse response the widening can also be used to synthesise diffuse early reflections.

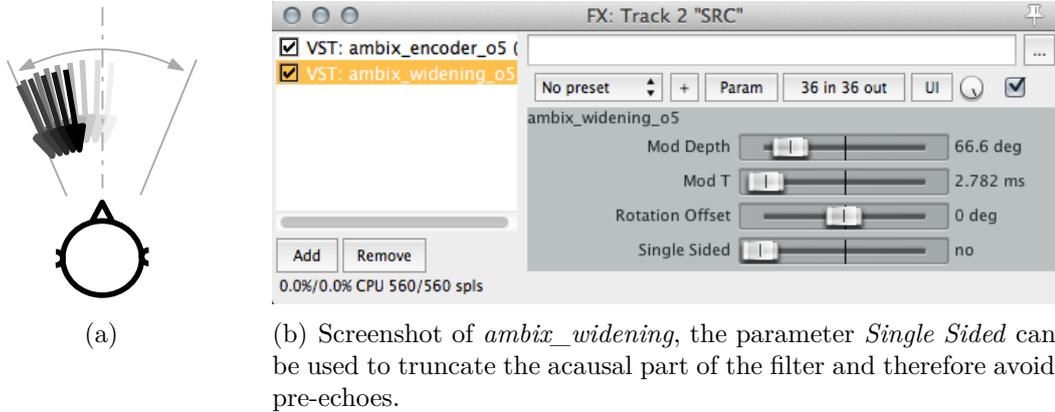
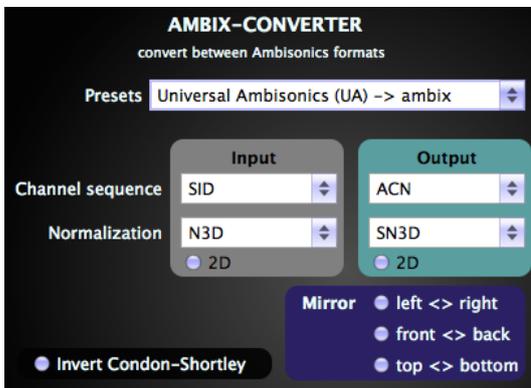


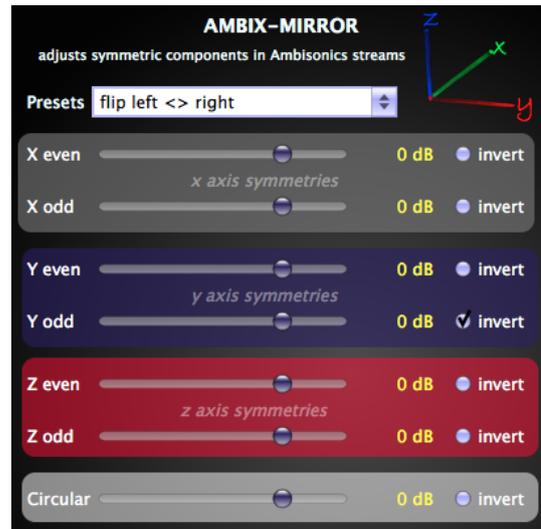
Figure 4.3: Frequency-dispersed direction of arrival as Ambisonic widening effect [ZFKC14].

Converting between Different Ambisonic Conventions

As already mentioned in Sec. 2.3.1 Ambisonics, suffers from different existing conventions regarding the sequence and weighting of the spherical harmonic components. To retain compatibility to software and recordings that are using different conventions *ambix_converter* (Fig. 4.4(a)) has been developed which is capable of converting between all coexisting Ambisonic conventions.



(a) *ambix_converter* can be used to incorporate recordings and software with different Ambisonic conventions within the same project.



(b) *ambix_mirror* can be used to correct for an incorrect surround image regarding the coordinate axes.

Figure 4.4: Plug-ins for the compensation of different Ambisonic conventions or mirrored coordinate systems.

Binaural Ambisonic Playback with Head-Tracking

The binaural decoder *ambix_binaural* (Fig. 4.6(b)) generates virtual loudspeaker signals that are convolved with corresponding *binaural room impulse responses* (BRIRs). The software includes ready-to-use binaural decoder presets using the measured impulse responses from venues with 24 up to 46 loudspeakers, namely from the IEM-CUBE (cf. Fig. 1.1), Mumuth Graz and SARC Belfast [HR13]. This allows sound artists and computer musician to prepare their surround content before they enter the actual venue and safe rehearsal time. Recording and mixing engineers can work on their mix before booking surround-equipped studios and consumers can enjoy surround sound without extensive loudspeaker setups.

Each virtual loudspeaker signal v_i has to be convolved with a stereo BRIR, resulting in 2L convolutions. The BRIRs of non-anechoic rooms can be quite long, depending on the reverberation time of the room. To reduce the computational load, the BRIRs have been divided into an early and a late part. The early part of the BRIRs is used for convolution with the individual loudspeaker signals. The late parts of all BRIRs are summed and fed from the sum of all virtual loudspeaker signals (cf. Fig. 4.5). Therefore the long part of the impulse response is not processed for each loudspeaker individually, resulting in a significant reduction of the computational load. The *mixing time* of a room impulse response is defined as the moment when the diffuse reverberation tail starts and depends on the geometry of the room. A detailed evaluation of predicting the mixing time can be found in [LKW10]. In informal listening sessions the cross-fade time was chosen for each set of BRIRs individually. The constant power cross-fades are performed approximately after the second order wall reflections.

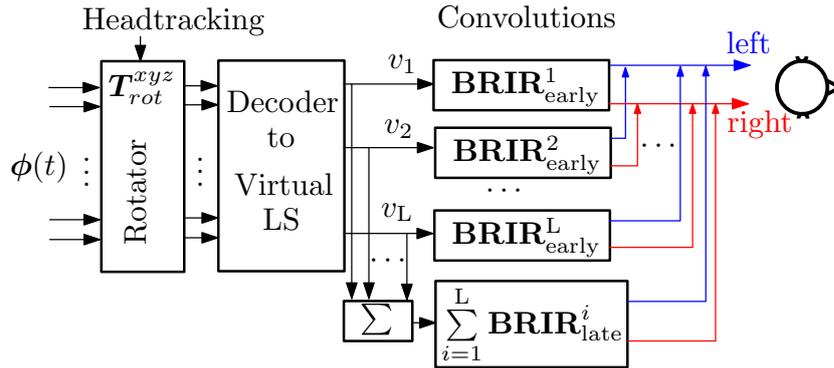


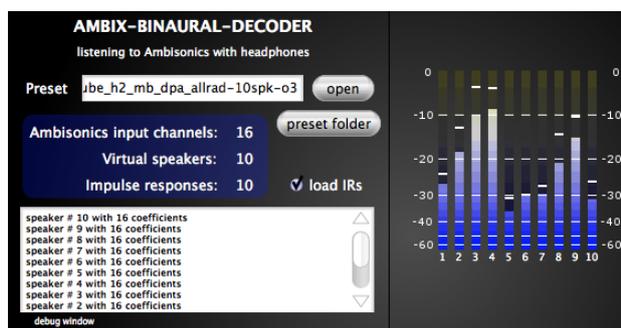
Figure 4.5: Dividing early and late part of the BRIRs to reduce computational load, the length of the early part is chosen depending on the dimensions of the room.

In combination with a head tracking system and the rotation plug-in *ambix_rotator*, an even more realistic simulation can be achieved by taking into account the head movements of the listener. A former publication by the author [Kro13] describes an optical head-tracking method using the Microsoft Kinect camera. This has the advantage of avoiding to have the listener wear any sensor, but the head pose estimation from a video signal adds substantial computational load. In order to save these resources for performing the numerous convolutions, a head tracking system based on a micro controller and sensor board can be used (cf. Fig. 4.6(a)) [Fri12]. The data from a 3-axis-accelerometer, gyro and magnetometer is used to determine the head rotation of the listener. All computations are performed on an Arduino Nano and the serial data is sent via USB or Bluetooth to the host computer. A

Pure Data patch receives the serial data and redirects the three rotation angles via OSC to the Ambisonic rotation plug-in *ambix_rotator*. The current implementation allows an update rate of around 60Hz. The total latency of the system is composed of the tracker update rate and the audio buffer size, which should be chosen as small as possible for fast reactivity. However FFT convolution gets more efficient with a larger block size, therefore a compromise might be necessary.



(a) Do-it-yourself drift-compensated headtracker composed of an Arduino Nano and 3-axis-accelerometer, gyro and magnetometer sensorboard.



(b) GUI of the binaural decoder showing the virtual loudspeaker levels, different presets with decoder matrix and sets of BRIRs can be loaded.

Figure 4.6: Binaural playback of Ambisonic recordings using *ambix_binaural*.

4.2 mcfx Audio Plug-ins

This section describes the multichannel effect plug-ins developed as toolbox for processing signals from loudspeaker and microphone arrays. The maximum number of channels for the *mcfx* plug-ins can be chosen at compile time and is restricted only by the capabilities of the host application. The following section will outline some classical as well as new areas of application for these multichannel plug-ins.

Equalization

mcfx_filter offers a high/low pass, high/low shelf and two peak filters. The filter can be used to modify the sonic characteristic of microphone arrays, Ambisonic recordings or as phase-aligned Linkwitz-Riley multichannel crossover. (Fig. 4.7, 4.9)

Gain and Delay Compensation

The *mcfx_gain_delay* plug-in can be used to compensate for time alignment or adjust the individual levels within a loudspeaker array (Fig. 4.7). Each channel passing the plug-in can be independently scaled and delayed.

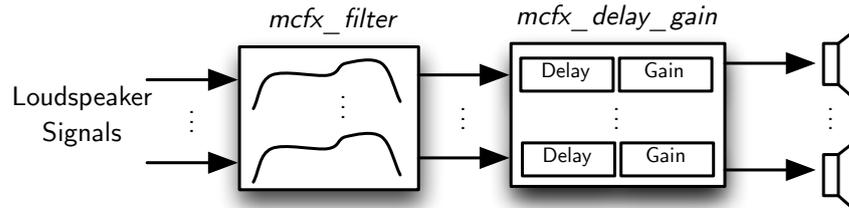


Figure 4.7: Using the *mcfx* plug-ins for individual loudspeaker equalization.

Convolution Matrix

The *mcfx_convolver* plug-in is a multichannel convolution matrix that uses an efficient open-source partitioned FFT convolution algorithm [HiF13]. An arbitrary number of impulse responses can be loaded and assigned to specific inputs and outputs. The convolver plug-in is currently used to drive an icosahedral loudspeaker array from 16 Ambisonic channels [Zot09b] (Fig. 4.8) to encode the signals of a 32 channel microphone array into 25 Ambisonic channels [Lös13] (Fig. 4.9), and to add Ambisonic reverb to recordings.

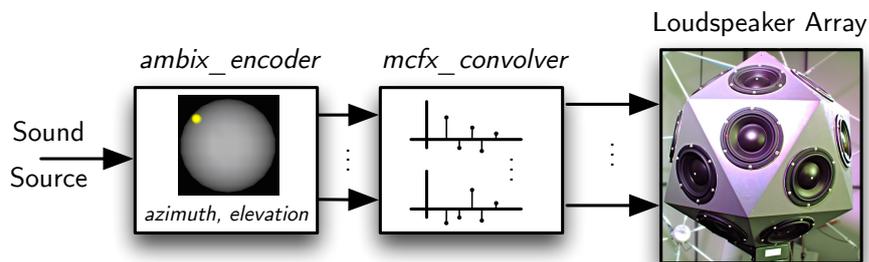


Figure 4.8: Using the encoder and convolver to control the sound radiation of the IEM icosahedral loudspeaker.

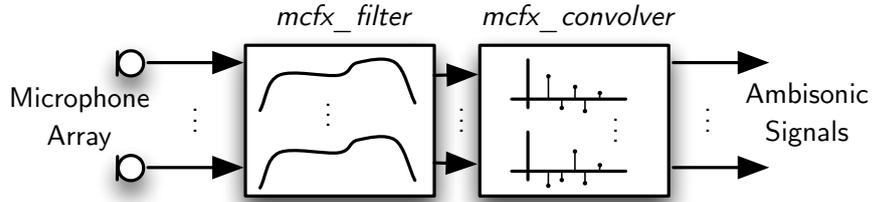


Figure 4.9: Adjusting the sonic characteristic of a microphone array and encoding to Ambisonics.

4.3 Metering Ambisonic Signals

Sound engineers are accustomed to being given visual feedback monitoring the common signal manipulations such as equalization, gain adjustments or panorama. Metering helps to verify the intended manipulation and therefore is desirable for Ambisonic recordings as well.

Fig. 4.10 shows the block diagram of a directional loudness level meter for Ambisonics that was developed in order to verify the manipulations described in this thesis. A *Pure Data*² prototype was developed using the graphical extension library *Gem* and the matrix calculation library *iemmatrix*.

The process of generating the visual surround image is divided in three parts, one operating in audio sample rate, one in control rate which defines the display frame rate and the final graphical rendering that is done on the graphics card (Fig. 4.10) using OpenGL³. The Ambisonic signal $\phi(t)$ is sampled using a t -design node configuration. The rms and peak value is measured for all L directional audio signals $f(\theta_l, t)$. High-pass and low-pass filtering may be applied to the input of the rms/peak detectors to have a frequency-selective display. The rms/peak detectors output their logarithmic measurements in the final display frame rate. For generating an interpolated texture these $2L$ control rate values are re-encoded into the spherical harmonic domain. A texture with a finer resolution of v vertical and h horizontal pixel is used to display the directional rms value. The color of the $P = hv$ pixel is determined by sampling the rms spherical harmonic vector using a subdivision of h for the azimuth and v for the elevation. Afterwards this texture is mapped onto a sphere. The peak values are sampled in lower spatial resolution and displayed as small spheres located as grid on

²Pure Data is a graphical programming environment for computer music and multimedia applications.

³OpenGL is an application programming interface for rendering 2D and 3D vector graphics.

the surface of the sphere. Additionally to the 3D view, the Mollweide projection is used to map the whole surround image onto a 2D image.

To reduce the CPU load, it might be feasible to move all processing for the metering to the graphic card.

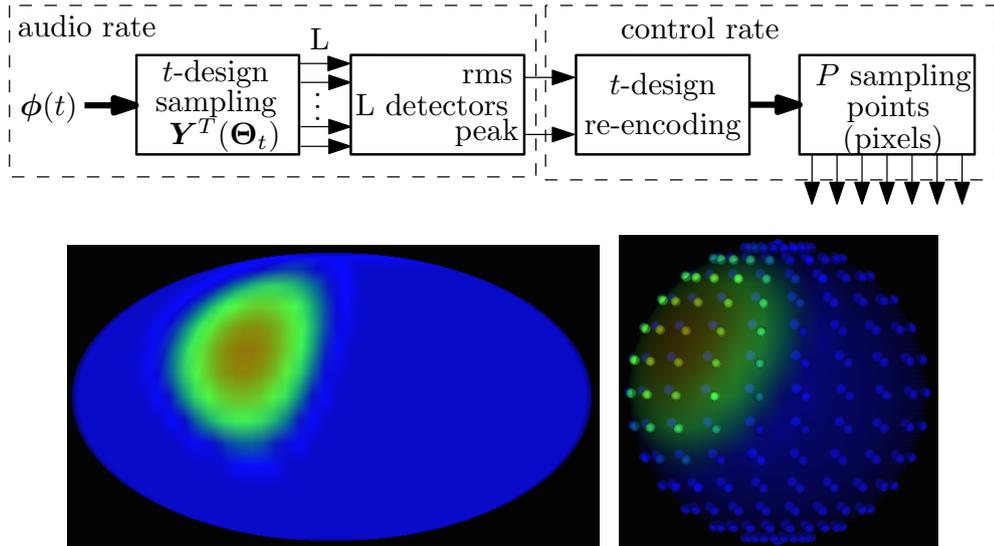


Figure 4.10: Directional loudness level meter for Ambisonics. The texture displays directional rms values, the small spheres obtain their color from the directional peak value. The peak values from the back side of the sphere are visible through the semitransparent texture.

4.4 Loudspeaker Installations

The following section lists several loudspeaker installations in which the software from this thesis was used to produce and play back Ambisonic content. Informal listening sessions reported stable localization in big parts of the listening area in all the setups. Using the AllRAD approach [ZF12] allowed to compute decoder matrices within seconds for different Ambisonic playback orders. This collection of photos and loudspeaker layouts should demonstrate the versatility of the Ambisonic tools established and gathered within this thesis.

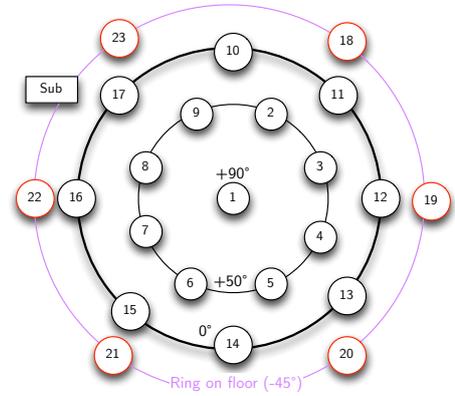


Figure 4.11: Vilnius University Astronomical Observatory, 2013. Low budget installation consisting of 23 loudspeakers in an observatory dome. In combination with a powerful subwoofer, it became clear that low-cost solutions are feasible and usable for creative experiments and performing concerts. Loudspeakers on the floor extended the surround coverage below the equator. The dome accommodates about 30 listeners.

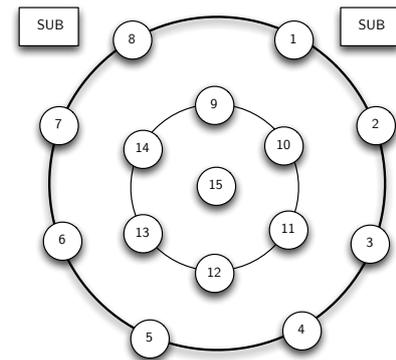


Figure 4.12: ICT 2013 conference, Litexpo, Vilnius, 2013. Larger scale installation with 15 Dynacord D-Lite D8 loudspeakers covering a 15x15m area in a crowded exhibition space. Public installations are problematic in terms of the required minimum loudspeaker height which in this case was 2.5m. Therefore it was not possible to place a sound source at the level of the listener.

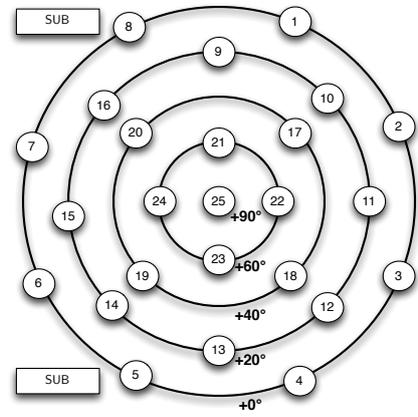


Figure 4.13: EAA Symposium Berlin, 2014. Mobile IEM Ambisonic installation, 25 Genelec 8020A loudspeakers in 5 elevation levels. After setting up, the individual loudspeakers have been adjusted by ear regarding loudness differences and bass response. This manual adjustments significantly improved the sonic and spatial performance of the installation. [Photo: ©FG Audiokommunikation, TU Berlin]

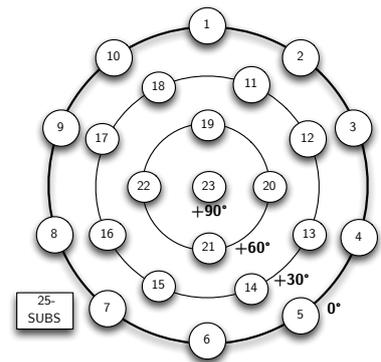


Figure 4.14: Music Innovation Studies Centre, Vilnius, 2014. Fixed installation of 23 Dynacord D-Lite D8 loudspeakers in the Lithuanian Music and Theatre Academy. At the time of writing, this installation has just been finished, and careful listening will be done in future.

Chapter 5

Conclusion and Outlook

This thesis started with a brief introduction in the representation of surround signals using Ambisonics. It has been shown that modification of the spatial image of Ambisonic recordings is a useful tool for extending the usability of surround sound in future applications. The general transformation approach from chapter 3 makes it possible to derive matrices for performing other thinkable directional manipulations. Several transformations require higher re-expansion orders and therefore have to be used with care. However, even with truncating the re-expansion order, the transformation might still yield the desired effect. Chapter 4 introduced the implementation of the derived algorithms as plug-ins. This suite of audio effects can be used to produce surround recordings in higher-order Ambisonics within digital audio workstations on all major operating systems.

Outlook

Several of the implemented plug-ins currently lack a graphical user interface. A simplified usage of the software is desirable targeting non-experts in Ambisonics. Hardware interfaces for the spatial mix of sound objects as well as controllers for the spatial manipulators would complement the usability of the software. Furthermore, effects such as directional compression and limiting would be desirable for mastering Ambisonic recordings. The topic *exchangeable Ambisonic data format* as missing link between producer and consumer has to be revisited and implemented for the practical use.

Bibliography

- [Ber12] B. Bernschütz, “Map Projections for the Graphical Representation of Spherical Measurement Data,” in *38th German Annual Conference on Acoustics*, Darmstadt, 2012.
- [CC09] M. Chapman and P. Cotterell, “Towards a comprehensive account of valid Ambisonic Transformations,” in *1st Ambisonics Symposium*, Graz, 2009.
- [Cha09a] M. Chapman, “A Standard for Interchange of Ambisonic Signal Sets,” in *1st Ambisonics Symposium*, Graz, 2009.
- [Cha09b] ———, “Symmetries of Spherical Harmonics: applications to ambisonics,” in *1st Ambisonics Symposium*, Graz, 2009.
- [Coc14] Cockos Inc. (2014) Reaper. [Online]. Available: <http://www.reaper.fm>
- [Cyc14] Cycling '74. (2014) MaxMSP. [Online]. Available: <http://cycling74.com>
- [Dav14a] P. Davis. (2014) Ardour. [Online]. Available: <http://ardour.org>
- [Dav14b] ———. (2014) JACK Audio Connection Kit. [Online]. Available: <http://jackaudio.org>
- [DH94] J. Driscoll and D. Healy, “Computing Fourier Transforms and Convolutions on the 2-Sphere,” *Adv. Appl. Math.*, vol. 15, no. 2, pp. 202–250, Jun. 1994.
- [EJZ14] N. Epain, C. Jin, and F. Zotter, “Ambisonic Decoding with Constant Angular Spread,” *EAA Symposium on Auralization and Ambisonics*, 2014.
- [Fri12] D. Frie. (2012) open-headtracker. [Online]. Available: <http://code.google.com/p/open-headtracker>
- [GB92] M. Gerzon and G. Barton, “Ambisonic Decoders for HDTV,” *92nd AES Convention, Vienna*, 1992.

- [Ger73] M. Gerzon, “Periphony: With-Height Sound Reproduction,” *J. Audio Eng. Soc.*, vol. 21, no. 1, pp. 2–10, 1973.
- [GJ⁺10] G. Guennebaud, B. Jacob *et al.* (2010) Eigen v3. [Online]. Available: <http://eigen.tuxfamily.org>
- [GP11] M. Gräf and D. Potts, “On the computation of spherical designs by a new optimization approach based on fast spherical Fourier transforms,” in *Numerische Mathematik Vol. 119 No. 4, p. 699-724*, 2011.
- [HB14] A. Heller and E. Benjamin, “The Ambisonic Decoder Toolbox,” in *Linux Audio Conference*, Karlsruhe, 2014.
- [HBL12] A. Heller, E. Benjamin, and R. Lee, “A Toolkit for the Design of Ambisonic Decoders,” in *Linux Audio Conference*, Stanford, 2012.
- [HiF13] HiFi-LoFi. (2013) FFTConvolver. [Online]. Available: <https://github.com/HiFi-LoFi/FFTConvolver>
- [HR13] F. Hollerweger and M. Rumori, “Production and Application of Room Impulse Responses for Multichannel Setups using FLOSS Tools,” in *Linux Audio Conference*, Graz, 2013.
- [HS96] R. Hardin and N. Sloane, “McLaren’s Improved Snub Cube and Other New Spherical Designs in Three Dimensions,” in *Discrete Computational Geometry, vol. 15, pp. 429-441*, 1996.
- [KKP09] J. Keiner, S. Kunis, and D. Potts, “Using NFFT 3 - a software library for various nonequispaced fast Fourier transforms,” in *ACM Trans. Math. Software, 36, Article 19, 1-30*, 2009.
- [KL80] V. Klema and A. Laub, “The singular value decomposition: Its computation and some applications,” *IEEE Transactions on Automatic Control, vol. 25, no. 2, pp. 164-176*, 1980.
- [Kro13] M. Kronlachner, “Ambisonics plug-in suite for production and performance usage,” in *Linux Audio Conference*, Graz, 2013.
- [LKW10] A. Lindau, L. Kosanke, and S. Weinzierl, “Perceptual Evaluation of Physical Predictors of the Mixing Time in Binaural Room Impulse Responses,” in *Audio Engineering Society Convention 128*, London, 2010.

- [Lös13] S. Lösler, “Schallfeldspezische Entzerrung bei Radialfiltern begrenzter Dynamik für das Eigenmike,” *Project thesis, University of Music and Performing Arts, Graz*, 2013.
- [Mal03] D. Malham. (2003) Higher order Ambisonic systems. [Online]. Available: http://www.york.ac.uk/inst/mustech/3d_audio/higher_order_ambisonics.pdf
- [NZDS11] C. Nachbar, F. Zotter, E. Deleflie, and A. Sontacchi, “ambiX - A Suggested Ambisonics Format,” in *Ambisonics Symposium*, Lexington, 2011.
- [Ple09] P. Plessas, “Rigid Sphere Microphone Arrays for Spatial Recording and Holography,” Master’s thesis, University of Music and Performing Arts, Graz, 2009.
- [Plo14] Plogue Art et Technologie, Inc. (2014) Bidule. [Online]. Available: <http://www.plogue.com>
- [PPP12] A. Politis, T. Pihlajamäki, and V. Pulkki, “Parametric Spatial Audio Effects,” in *15th Int. Conference on Digital Audio Effects*, York, 2012.
- [Pul97] V. Pulkki, “Virtual sound source positioning using vector base amplitude panning,” in *J. Audio Eng. Soc., Vol. 45, No. 6*, 1997.
- [Pul06] —, “Directional audio coding in spatial sound reproduction and stereo upmixing,” in *AES 28th International Conference*, Piteå, 2006.
- [PZ09] H. Pomberger and F. Zotter, “An Ambisonics Format for Flexible Playback Layouts,” in *1st Ambisonics Symposium*, Graz, 2009.
- [PZ11] —, “Warping of 3D Ambisonic Recordings,” *Ambisonics Symposium, Lexington*, 2011.
- [Raf05] B. Rafaely, “Analysis and Design of Spherical Microphone Arrays,” in *IEEE Transactions on Speech and Audio Processing, Vol. 13, No. 1*, 2005.
- [SF06] F. Simons and F. Dahlen, “Spherical Slepian functions and the polar gap in geodesy,” *Geophysical Journal International, vol. 166, no. 3, pp. 1039-1061*, 2006.
- [Son03] A. Sontacchi, “Dreidimensionale Schallfeldreproduktion für Lautsprecher- und Kopfhöreranwendungen,” Ph.D. thesis, Technical University Graz, Graz, 2003.

- [SS11] H. Sun and P. Svensson, “Design 3-D High Order Ambisonics Encoding Matrices Using Convex Optimization,” in *Audio Engineering Society Convention 130*, May 2011.
- [Sto14] J. Storer. (2014) JUCE (Jules’ Utility Class Extensions). [Online]. Available: <http://www.juce.com>
- [SW09] I. Sloan and R. Womersley, “A variational characterisation of spherical designs,” *J. of Approx. Theory* 159:308–318, 2009.
- [Wei14a] E. Weisstein. (2014, May) Frobenius Norm. [Online]. Available: <http://mathworld.wolfram.com/FrobeniusNorm.html>
- [Wei14b] ——. (2014, May) Rotation Matrix. [Online]. Available: <http://mathworld.wolfram.com/RotationMatrix.html>
- [Wil99] E. Williams, *Fourier Acoustics*. Academic Press, 1999.
- [ZF12] F. Zotter and M. Frank, “All-Round Ambisonic Panning and Decoding,” in *J. Audio Eng. Soc., Vol. 60, No. 10*, 2012.
- [ZFKC14] F. Zotter, M. Frank, M. Kronlachner, and J.-W. Choi, “Efficient Phantom Source Widening and Diffuseness in Ambisonics,” in *EAA Symposium on Auralization and Ambisonics*, Berlin, 2014.
- [Zm02] J. Zmólnig, “Entwurf und Implementierung einer Mehrkanal-Beschallungsanlage,” Master’s thesis, University of Music and Performing Arts, Graz, 2002.
- [Zot09a] F. Zotter, “Sampling Strategies for Acoustic Holography/Holophony on the Sphere,” in *NAG-DAGA*, 2009.
- [Zot09b] ———, “Analysis and Synthesis of Sound-Radiation with Spherical Arrays,” Ph.D. dissertation, University of Music and Performing Arts, Graz, 2009.
- [ZP11] F. Zotter and H. Pomberger, “Warping of the Recording Angle in Ambisonics,” *1st International Conference on Spatial Audio, Detmold*, 2011.
- [ZP12] ———, “Spherical Slepian functions for approximation of spherical measurement data,” in *Fortschritte der Akustik - DAGA*, 2012.
- [ZPN12] F. Zotter, H. Pomberger, and M. Noisternig, “Energy-Preserving Ambisonic Decoding,” *Acta Acustica United with Acustica, Special Issue on Spherical Acoustics pp. 37-47(11), vol. 98(1)*, 2012.