AMBISONICS SYMPOSIUM 2011
June 2-3, Lexington, KY

AMBIX - A SUGGESTED AMBISONICS FORMAT

Christian Nachbar[1], Franz Zotter[2], Etienne Deleflie[3], Alois Sontacchi[2]

[1] Student, University of Music and Performing Arts, Graz (christian.nachbar@student.kug.ac.at)
[2] University of Music and Performing Arts, Graz (zotter@iem.at, sontacchi@iem.at)
[3] University of Wollongong, Australia (edeleflie@gmail.com)

Abstract:
Ambisonics is a 3D audio surround rendering and representation approach based on spherical harmonics with loudspeaker independent transmission channels. Although it was developed in the seventies and the techniques are well known, there are disagreements how to normalize, store and exchange Ambisonic data.
This paper's mission is to propose a standard for the Ambisonics community and it should be seen as an encouragement to use the proposed convention, in order to facilitate exchange and communication.
This standard is named ambiX (Ambisonics exchangeable).
For the case that mixed order or reduced Ambisonic signal sets are beneficial, the format includes a simple matrix - named "adaptor matrix". This matrix can be freely configured to re-order, complete, re-normalize, or embed the transmitted or stored audio channels to a default set of periphonic Ambisonic signals. It is advocated as extended *format, because it enables new techniques to reduce the Ambisonic signal set utilizing prior knowledge about restricted surround geometries like hemispherical or circular. Providing a default periphonic decoder for all surround geometries ensures universal applicability, while inspection tools for the adaptor matrix are advanced means for optimal decoding.*
IEM provides a small "proof of concept" implementation of tools to write, read audio data and the adaptor matrix, as well tools to analyze the adaptor matrix graphically to select the best decoder for given files.

Key words: Ambisonics, playback on partial domains, file format, signal set format, interchange, standard, data format, data exchange, meta data, archiving, storage, playback

1 INTRODUCTION

Imagine you are at a famous venue. Soon you'll premiére with your newest Ambisonic composition and there are only a few minutes left to prepare your installation and get your piece of music to work. And within moments of anticipation your optimism is interrupted by the responsible audio engineer who is not familiar with the strange channel ordering and normalization of your recording. Even the file format of your work seems unrecognizable by the available hardware.

Picture yourself the outcome of this situation: There might be a solution or not. And this is what this paper is about: **ambiX**, a flexible specification for Ambisonic recordings in order to compose, record, exchange and play back them easily without concerning about incompatibility. A good file format cannot absolutely guarantee a successful performance, but it might be a good basis to avoid at least technical problems.

This paper will introduce actual formats and software tools and will refer to the need for an agreement to a standard format. After discussing previous standardization attempts a new format specification including a technique to compress data in mixed order situations will be proposed.

B-format: The ".amb" Format, FuMa. Frankly, the well known B-format could be seen as the only *de-facto* existing Ambisonic format – and the most wide-spread hardware and software tools are based on this – but due to its restrictions regarding number of channels and file size, it needs to be reconsidered. A good overview of the formats is provided on the website of Martin Leese [1] in terms of the Microsoft's Wave Format Extensible, using the extension amb. The B-format allows mixed order systems up to horizontal oder 3 and height order 3 (16 channels). The B-format use the Furse-Malham [2] (FuMa) set of weighting factors. The biggest drawback is the file-size limit of 4GB due to the Microsoft WAV-format as the data container

1.1. Available Software

Furthermore, there are various software packages supporting Ambisonics using different conventions. The list is a result of research on the internet, therefore might not be complete, but it gives a good overview:

SuperCollider[*1] **:** "UGen"[*2] plugins by Joshua Parmenter *BFEncode1*, *BFEncode2*, *BFDecode1*, *BFMixer, BFManipulate* for 1st order B-Format; *FHMEncode1, FHMEncode2, FHMDecode1* for 2nd order Furse-Malham (FuMa) set.

MAX[*3] **:** A set of externals[*4] for MAX for Ambisonics surround sound processing and source-control by Philippe Kocher and Jan Schacher (Institute for Computer Music and Sound Technology). The objects `ambiencode` and `ambidecode` expect B-format signals.

Plogue Bidule[*5] **:** *Ambisonic Bidules*[*6] by Aristotel Digenis, a Bidule plugin suite based on his Ambisonic C++ Library *amblib*[*7] which can handle Ambisonic signals in B-format up to order 3.

Ardour[*8] **:** Fons Adriaensen and Jörn Nettingsmeier developed *AMB-plugins*[*9] for the LADSPA plugin format, which uses FuMa.

Pure Data (Pd)[*10] **:** IEM also provides some applications using Ambisonics – mainly in Pd – including *bin_ambi*[*11] [3], a Pd application for binaural rendering using Ambisonics, *iem_ambi*[*12] , a Pd Library to calculate Ambisonic encoder matrices, rotation matrices and decoder matrices up to 4th order, and *CUBEmixer*[*13] , a real-time mixing and mastering for multichannel speaker systems or binaural audio with integrated room simulation.

LINUX native: Fons Adriaensen also created *AmbDec*[*14] , an Ambisonic decoder, which handles FuMa as well as N3D and SN3D [4]. Jörn Nettingsmeier describes his practical experience using AmbDec in [5].

VST[1] **plugins:** *B-pan* and *B-dec*[*15] by Dave Malham for 1st order B-format. *B2X Plug-In Suite for Mac OS X*[*16] by Daniel Courville for 1st order B-format, 2nd order and 5th order planar B-format with FuMa.

Ambisonics related conventions as well as techniques have been developed by institutions and persons who believed in the benefits of this approach which apparently has not been appealing the broad mainstream of pair- or triplet-wise stereophony users. The pioneers in Ambisonics, who accepted the mathematical and technical challenge of improving and extending the principle, have been arriving at various individual custom built solutions, which have all been built to sound good and fulfill individual practical constraints.

1.2. Where are we now?

Nowadays, beginners in Ambisonics do not need to be pioneers and can use various publications to obtain an understanding of Ambisonics, and they can even choose one of the software packages listed above that suits their application. However, it might seem confusing to regard the amount of different proposals and software packages that use different limitations and practical conventions, and not all of the packages are well documented. Unified standards can simplify for beginners, programmers, and users to obtain similar results and possibilities in different environments. The past discussions and the past evolution showed that the evidence and inclusion of all contributors in the process of agreement is necessary not only to arrive at a broad consensus, but also to recognize which arguments and technical requirements are the most most relevant ones.

1.3. Standardization attempts

Only recently, there is great interest in harmonizing the confusing number of different solutions. This is probably inspired by the increased research effort, the various open source toolboxes, the development of higher order techniques, and spherical microphone arrays.

Daniel and MPEG-4. Jérôme Daniel wrote the earliest and most comprehensive work on higher order Ambisonics with his thesis [6]. In the later paper [7] he proposed to include distance coding into an Ambisonic format, preferably within an MPEG-4 audio stream description. The amendment [8] based on his work describes the a rigorous attempt of implementing all normalizations and possibilities of such an Ambisonic format, including distance parameters, diffuseness, various normalizations, a recombination matrix for adaptation of the audio channels to Ambisonics, coordinate definitions, and options on the playback facility (headphones/loudspeakers). However, it seems the huge effort in this standardization document has not been adopted by the small community producing and using Ambisonics with their own tools.

Discussion meeting at IEM. In 2008, a small group met at IEM in Graz for discussion[2] with people from Zürich and Parma to discuss new possibilities of other signal sets for hemispherical, symmetrical, or circular signals, and the necessity of including distance information.

Ambisonics Symposium 2009, ambisonics.ch, and Chapman. Michael Chapman [9] proposed at the first Ambisonics Symposium in 2009 a file-format using the Core Audio Format container [10], the Ambisonics channel number sequence (ACN [11]), N3D normalisation, and additional meta-data as XML tree.

The discussions in a bigger community mainly revealed that a broad consensus required more time and discussion[3].

Universal Ambisonic. A link to its specification can be found at [12]. At the moment the latest version of the Universal Ambisonic specification is 0.983, and it supports the orders from first up to forth, N3D, the "Gerzon Ambisonic" sequence., cf. [6], mixed orders by empty channels, lossless audio compression with Wavpack. As it is now, the website `http://soundofspace.com` maintained by Etienne Deleflie offers about 30 recordings in Universal Ambisonic.

New mixed order sets. Chris Travis [13] presented at the Ambisonics Symposium in 2009 a channel scheme for mixed order signal sets which *"a way forward that is free of the limitations of horizontal-only systems yet relatively unencumbered by the costs and difficulties of conventional fully-periphonic systems"*. Moreover, Travis provided in-

[1]Virtual Studio Technology: an audio processing plugin-format by Steinberg

[2]`http://ambisonics.iem.at/xchange/meeting08/ambixchange-discussion-minutes`

[3]`http://ambisonics.iem.at/symposium2009/proceedings/format-discussion` (live recording)

teresting material for a lively discussion on normalization schemes.

1.4. Where are the disagreements?

The attempt to write about a conceivable Ambisonic standard feels like entering a minefield of terminology. Ambisonics is based on spherical Harmonics (SHs) which are applied in many disciplines, and all of them have their own policy to normalize or even name them. And as a result, also within the young Ambisonics community disunity is present.

Four main discrepancies remain from the previous standardization attempts: normalization, ordering, the data container, and the correct nomenclature (is the azimuth harmonic indexed with order, degree, or function index?, there are even different letters for the harmonics: Y_n^m, Y_{lm}, y_n^m, S_{lm}, R_{lm}, $\ldots$). Although the common used normalization scheme is N3D, SN3D is easier to use when dealing with audio data: clipping of the higher order signals can be avoided, because the peak amplitude of single point sources will never exceed the level of the 0^{th} order signal; in practice this turns out to be more relevant than the N3D normalization for statistical diffuse fields.

Despite the above disharmony, the Ambisonics community agrees not to use the Condon-Shortley phase $(-1)^m$, because it causes confusion (it simplifies things only in quantum mechanics [14]).

As the FuMa letter based nomenclature is practically limited to 3rd or 4th order, SHs have to be ordered in a more flexible way. Concerning this unpleasantness, an overview of papers and books from various other disciplines has been searched for this paper. Nevertheless, a clearly defined channel sequence was only found in computer graphics, where SHs are usually indexed by $i = (l+1)\,l+m$ [15, 16]. Records of *ordered* SHs are only sporadic in geodesy [17], mathematics [18], crystallography [19], chemistry [20], physics [21] and quantum mechanics [14, 22]. However, there frequently is no need of arranging the set of SHs hence no consistent sequence. In fact, it only needs to be defined for the implementation of a numerical calculation. Speaking of computer graphics, Direc3D – part of Microsoft's DirectX application programming interface – has also an implementation of SHs [23] for 3D lighting using the index $i = (l+1)\,l+m$; this is equivalent to ACN and used here to specify ambiX . There are eminently readable books [24, 25, 26, 27, 28] about the basics concerning SHs whose math does not rely on any SHs sorting convention.

1.5. Organization of contents

Regarding the various options, the authors and their discussion partners have been preparing solutions that appeared to be simple (*basic* format) and extensible (*extended* format) keeping the complexity as low as possible.

The second and third sections describe the *basic* format, its sorting and normalization convention, the resulting set of Ambisonic signals in the angular space, and its data storage formats. Section four describes the *extended* format which uses an adaptor matrix and several application examples of this matrix. For instance, how to use it for converting different orders and normalizations, for embedding of horizontal only signals, mixed order content, and the recent developments for surround geometries that cover parts of the angular space only. Section five announces software tools whose development has been started during the work on this paper and will be maintained online.

2 AMBIX: FUNDAMENTALS

As Ambisonic assigns signals to different angular patterns exciting the incident sound field, useful definition requires to define how to sort and normalize the basic patterns that are used, namely the spherical harmonics, see fig. 4(a). As a coordinate system, it is useful to define the x-axis pointing into the main look direction of the listener, the y-axis as the left, and the z-axis as the top direction.

2.1. Spherical harmonics definition, indexing, and normalization

As definition for real spherical harmonics, the following scheme seems to provide an agreeable[4] definition

$$Y_n^m(\varphi,\vartheta) = N_n^{|m|}\,P_n^{|m|}(\sin(\vartheta)) \begin{cases}\sin(|m|\varphi), & \text{for } m<0\\ \cos(|m|\varphi), & \text{for } m\geq 0\end{cases} \quad (1)$$

using the indices n and m for the order and degree, respectively, with the ranges $0 \leq n \leq \mathrm{N}$, and $-n \leq m \leq n$. P_n^m are the Legendre-functions after their definition [29] and N_n^m is a normalization term for both the Legendre functions and the trigonometric functions. The angle φ is the azimuth angle starting at the frontal direction and running counterclockwise, ϑ is the elevation, which is zero at the horizontal plane and negative below.

To build the set of Ambisonic signals, the channels corresponding to the SHs are ordered by ACN[5]:

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

The normalization that seems most agreeable is SN3D:

$$N_n^{|m|} = \sqrt{\frac{2-\delta_m}{4\pi}\,\frac{(n-|m|)!}{(n+|m|)!}}\,, \quad (3)$$

[4]This seems to be agreeable as it only uses positive multipliers m and the Legendre-functions with identical scaling for both the cosine and sine cases with same $|m|$. The real-valued notation of the azimuth harmonics (trigonometric functions) seems to be preferable to a complex exponential formulation for several reasons. On the one hand, a real-valued multiplication is preferable in real-time signal processing and storage. On the other hand, the otherwise formulated frequency dependent phase-shift $e^{\mathrm{i}m\varphi+\mathrm{i}\omega t}$ is, in general, not meaningful for the description of sound fields. Despite the zenith angle $\pi/2-\vartheta$ is usually the standard coordinate, elevation is considered more intuitive for audio.

[5]It can be easily handled using two loops and as well allows easy evaluation of $n = \lfloor\sqrt{\mathrm{ACN}}\rfloor$ and $m = \mathrm{ACN} - n^2 - n$.

and it is required to be evaluated only for positive m; the Kronecker-delta δ_m is one for $m = 0$ and zero otherwise.

Ambisonic signal representation in the angular space. Using the spherical harmonics $Y_n^m(\varphi, \vartheta)$, or in with the new index $Y_{ACN}(\varphi, \vartheta)$, a set of Ambisonic audio signals x_{ACN} is expanded to a theoretically continuous driving signals of the sound field

$$x(\varphi, \vartheta, t) = \sum_{n=0}^{\mathrm{N}} \sum_{m=-n}^{n} Y_n^m(\varphi, \vartheta)\, x_{nm}(t) \tag{4}$$
$$= \sum_{ACN=0}^{(\mathrm{N}+1)^2-1} Y_{ACN}(\varphi, \vartheta)\, x_{ACN}(t).$$

For a discrete setup of loudspeakers, this driving function is either sampled directly at the loudspeaker locations [30], or mode-matching decoders are computed [31], in some cases distance coding is desirable [32].

2.2. Data container

The container for the Ambisonic audio data is the *Core Audio Format (CAF)* as specified by Apple [10]. Apple's CAF is a flexible file format with the advantages of practically unrestricted file size and number of channels

Besides the *CAF File Header*, the CAF format requires only two chunks: the *Audio Description Chunk* which holds fundamental audio information like the sample rate, the audio-coding format, the resolution and number of channels and the *Audio Data Chunk* holding the raw audio data (PCM).

In addition to the required chunks, the ambiX file format only utilizes the *User-Defined Chunk*, which we will use to store the adaptor matrix for the *extended* (sec. 4) ambiX file format.

There are many other chunks in the CAF file format specification, which will be gently ignored by our implementations. For instance the *Channel Layout Chunk* information assigning physical loudspeaker channels is useless in Ambisonics.

2.3. Optional lossless compression with WAVPACK

Although the option to compress the ambiX CAF file is not part of the ambiX specification, this is an interesting option. Empty channels (without dithering) are compressed efficiently and block-wise decompression for streamed playback is also supported. The header information (including the adaptor matrix) of the ambiX file is not compressed and can be accessed directly.

The current official Release of WavPack [33] (4.60.1) does not support CAF yet. There is a development branch of WavPack which supports CAF and seems to work fine. David Bryant, author of WavPack, announced to release this branch in the future, after finishing and careful testing. The custom uuid-chunk of the *extended* format (next section) can be easily read from the compressed WavPack file, because it contains all original uncompressed header information.

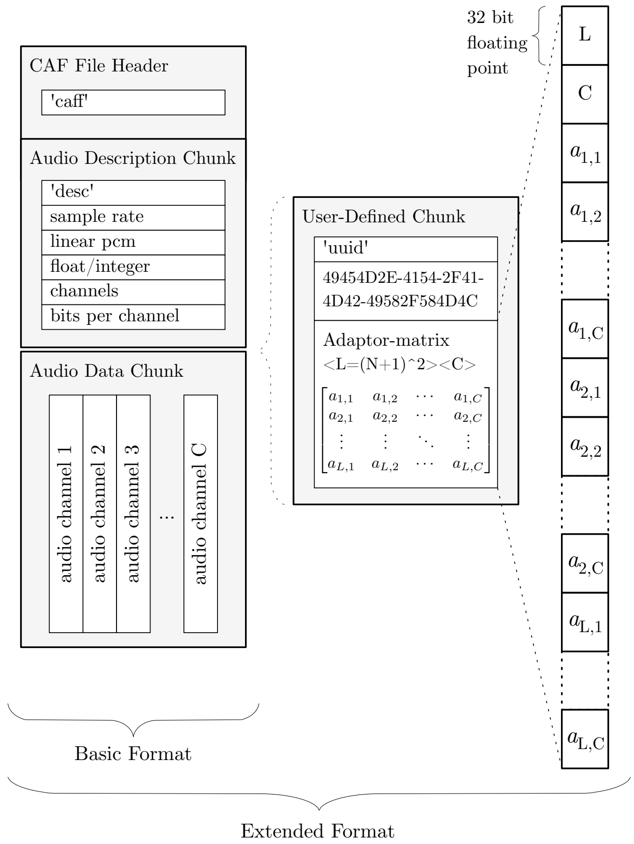


Figure 1: File layout of ambiX *basic* & *extended* format

3 AMBIX: BASIC FORMAT

ambiX specifies Linear PCM (pulse-code modulation) with samples of 16, 24 and 32 bit signed integer or 32 bit floating point with any valid sampling rate specified by CAF.

At the *basic* format, the channels represent N^{th}-order full periphony (3D). The number of channels allows to uniquely identify the order. **Mixed order** can be handled in two different ways: empty channels can be provided which are handled efficiently by optional WAVPACK compression (sec. 2.3) or, alternatively, by using the *extended* format as described in sec. 4.4.

3.1. No meta data

The ambiX *basic* format entirely avoids meta data to keep implementation and user efforts small. The advantage is that handling of the files is simple: hardware encoder/decoder handle audio channels without additional information, also wave-editing with audio processing plugins (VST, Audio Units) process the signals based on a determined channel sequence and normalization. Only the known tools are necessary for rendering of *basic* ambiX files.

4 AMBIX: EXTENDED FORMAT

If signals sets are beneficial which are not fully periphonic, the *extended* ambiX format is useful as it contains an adap-

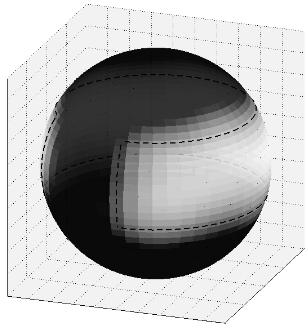


Figure 2: Energy fluctuation of panning, using a custom shaped adaptor matrix, N=4; the dotted line marks the border of the surround playback geometry.

tor matrix (fig. 3) in the *User-Defined Chunk* with the Universally Unique Identifier
(UUID) **49454D2E-4154-2F41-4D42-49582F584D4C**.

If this chunk is present, the channels cannot be directly played back as Ambisonic signals but need conversion by the provided matrix. In this case, the default set of periphonic output signals are defined by the output dimension of the matrix and not by the number of the stored channels.

Within the user-defined chunk, this matrix is represented as 32 bit floating point values in big-endian byte order. The first two numbers describe the dimension (rows = default periphonic output channels, columns = number of audio channels in file) of the matrix followed by the matrix cells ordered row by row.

If this chunk is not present, the file should be treated as ambiX *basic* format.

4.1. Adaptor Matrix

The adaptor matrix (fig. 3) restores a full periphony set of $L = (N+1)^2$ Ambisonic signals/channels from a physical set of $C \leq L$ stored signals/channels, see 3.

Before we go into technical details, let us assume a given adaptor matrix and visualize its target surround geometry on the sphere. This is done by analyzing the adaptor matrix in terms of its energy mapping, which can be done by utilizing *Parseval's theorem*. In fig. 2, important areas on the sphere are bright, while dark areas mark areas are suppressed in the sound material of the file. In the same way, hemisphere, horizontal only, or other partial signal sets can be distinguished easily.

4.2. Exemplary use of the adaptor matrix.

This section gives some examples on how the adaptor matrix can be employed to restore compatibility between different orders, normalizations, or basis systems with the proposed format.

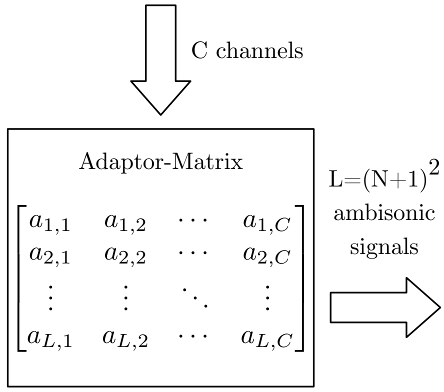


Figure 3: Adaptor matrix for signal reconstruction/adaption.

4.2.1 Converting from B-format.

In the simplest example, the order of the given audio material needs to be adapted. To *reorder* the channels from B-format sequence (WXYZ) to ACN sequence, the following matrix is used:

$$A_{(\mathrm{B}\rightarrow\mathrm{ACN})} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

and yields the signals

$$\begin{bmatrix} x_{\mathrm{ACN}_0} \\ x_{\mathrm{ACN}_1} \\ x_{\mathrm{ACN}_2} \\ x_{\mathrm{ACN}_3} \end{bmatrix} = A_{(\mathrm{B}\rightarrow\mathrm{ACN})} \begin{bmatrix} x_{\mathrm{B}_W} \\ x_{\mathrm{B}_X} \\ x_{\mathrm{B}_Y} \\ x_{\mathrm{B}_Z} \end{bmatrix}.$$

However usually, not only the order but also the weighting needs to be adjusted. With coefficients other than one, also *renormalization* is performed. Therefore, the matrix $A_{(\mathrm{B}\rightarrow\mathrm{ACN})}$ used to convert a B-format signal set into an ambiX compliant file finally is:

$$A_{(\mathrm{B}\rightarrow\mathrm{ambiX})} = \begin{bmatrix} \sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

4.3. Horizontal-only signals

Horizontal-only signals used for driving a horizontal ring of loudspeakers are normally given in terms of a circular harmonics decomposition, i.e. a decomposition into trigonometric functions $\sin(|m|\varphi)$ and $\cos(|m|\varphi)$

$$x(\varphi) = \sum_{m=-\mathrm{N}}^{\mathrm{N}} x_{\mathrm{hor},m}\, \Phi_m(\varphi), \tag{5}$$

$$\Phi_m(\varphi) = \sqrt{\frac{2-\delta_m}{2}} \begin{cases} \sin(|m|\varphi), & \text{for } m < 0 \\ \cos(|m|\varphi), & \text{for } m \leq 0 \end{cases}. \tag{6}$$

In order to re-encode this kind of horizontal-only Ambisonic signal set into the proposed format, it is expanded with a Dirac delta distribution $\delta(\sin(\vartheta))$, which embeds the circular signal at the horizon of a sphere, yielding a zero signal elsewhere. It is easy to see that the spherical harmonics contain the circular functions . By exploiting orthogonality and integration in the equation

$$\sum_{n,m} x_{nm}\, Y_n^m(\varphi,\vartheta) = \delta(\sin(\vartheta)) \sum_m x_{\mathrm{hor},m}\, \Phi_m(\varphi), \quad (7)$$

and the knowledge that $Y_n^m(\varphi,\vartheta) = \frac{N_n^{|m|}}{(2-\delta_m)} P_n^{|m|}(\cos(\varphi))\, \Phi_m(\varphi)$, horizontal only signals are converted with

$$x_{nm} = a_m^{nm}\, x_{\mathrm{hor},m}, \quad (8)$$

$$a_m^{nm} = \sqrt{\frac{(n-m)!}{(n+m)!}}\, P_n^m(0), \quad (9)$$

$$\boldsymbol{x}^{sph} = \boldsymbol{A}\, \boldsymbol{x}^{\mathrm{hor}}. \quad (10)$$

The related functions are shown in fig. 4(b).

4.4. Mixed order signals

Mixed order signals (#H#P, #H#V) [13] can also be coded into the adaptor matrix without the need of additional meta data. From a given adaptor matrix, known mixed-order schemes are easily detected due to their sparse structure. A 1H0V configuration, e.g., has a characteristic row of zeros

$$A_{(\mathrm{1H0V}\to\mathrm{ambiX})} = \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

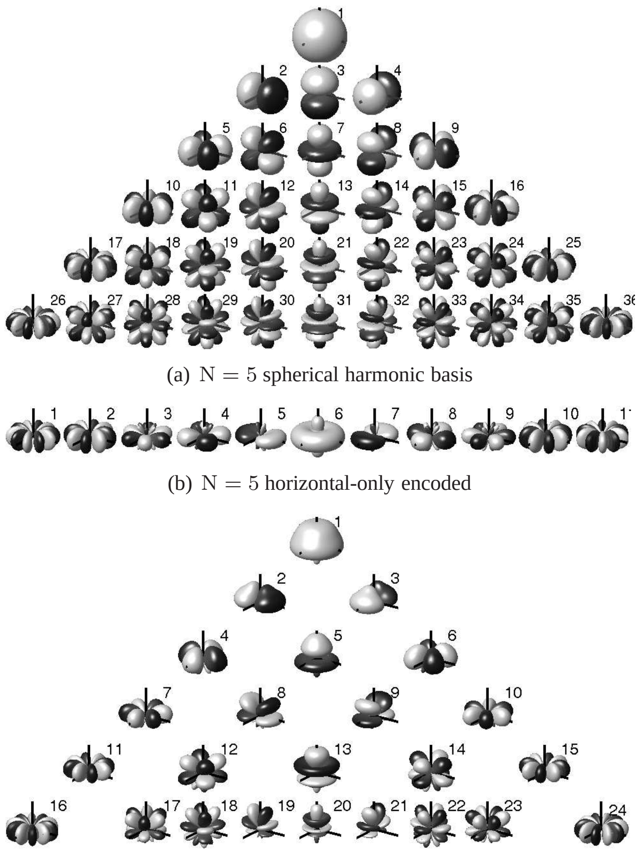
4.5. Custom shaped signals: eigenvalue decomposition

The papers [34, 35] have demonstrated that a smaller set of Ambisonic signals can be used if the target surround array is hemispherical, cf. 4(c), or only covers any other fraction of the sphere. Ideally, Ambisonic panning always yields a constant energy on the sphere for any panning direction. This is not the case when regarding the energy on only a fraction of the sphere.

For the above definition of the expansion coefficients $\tilde{x}_{nm}$ fulfill Parseval's theorem on the entire sphere $\mathbb{S}^2$ when using the orthogonality of the N3D normalized spherical harmonics $\tilde{Y}_n^m(\boldsymbol{\theta}) = \sqrt{2n+1}\, Y_n^m(\boldsymbol{\theta})$, i.e. $\int_{\mathbb{S}^2} \tilde{Y}_{n'}^{m'}(\boldsymbol{\theta}) \tilde{Y}_n^m(\boldsymbol{\theta})\, d\boldsymbol{\theta} = \delta_{n'n}^{m'm}$; the unit vector $\boldsymbol{\theta}$ is used to express the angular dependency on φ, ϑ:

$$E = \int_{\mathbb{S}^2} x^2(\boldsymbol{\theta})\, d\boldsymbol{\theta} = \sum_{n=0}^{\mathrm{N}} \sum_{m=-n}^{n} (2n+1)\, \tilde{x}_{nm}^2 \quad (11)$$

The trick of how to obtain a reduced set of functions is to determine the energy only over a fraction $S^2 \in \mathbb{S}^2$ of the sphere by the means of numerical integration [36]. Gathering the coefficients x_{nm} into a vector $\boldsymbol{x}$, the loss of the



(a) $\mathrm{N} = 5$ spherical harmonic basis

(b) $\mathrm{N} = 5$ horizontal-only encoded

(c) $\mathrm{N} = 5$ hemispherical basis

Figure 4: 36 spherical harmonics for $\mathrm{N} = 5$ (top) versus 11 basis functions for encoded horizontal-only and 24 basis functions for the hemisphere.

orthogonality can be expressed by

$$E = \int_{S^2} x^2(\boldsymbol{\theta})\, d\boldsymbol{\theta} = \tilde{\boldsymbol{x}}^{\mathrm{T}}\, \boldsymbol{G}\, \tilde{\boldsymbol{x}}. \quad (12)$$

Herein, the matrix $\boldsymbol{G}$ contains all the linear dependencies of one spherical harmonic with each other one on S^2. It is convenient to obtain its eigendecomposition $\boldsymbol{G} = \boldsymbol{V}\, \mathrm{diag}\{\boldsymbol{\lambda}\}\, \boldsymbol{V}^{\mathrm{T}}$. By defining $\tilde{\boldsymbol{x}}$ in terms of a new set of Ambisonic signals $\tilde{\boldsymbol{x}} = \boldsymbol{V}\, \boldsymbol{w}$, one observes simplification of $E = \boldsymbol{w}^{\mathrm{T}}\, \mathrm{diag}\{\boldsymbol{\lambda}\}\, \boldsymbol{w}$, and that there are only $\mathrm{K} \leq (\mathrm{N}+1)^2$ components with non-vanishing eigenvalues $\boldsymbol{\lambda}$ contributing to the energy. Using a correspondingly truncated system of eigenvectors $\boldsymbol{V}_{\mathrm{c}}$ of the dimensions $(\mathrm{N}+1) \times K$, the spherical harmonics $\boldsymbol{y}_{\mathrm{N}}(\boldsymbol{\theta})$ can be converted to a smaller set of functions $\tilde{\boldsymbol{v}}(\boldsymbol{\theta}) = \boldsymbol{V}_{\mathrm{c}}^{\mathrm{T}}\, \tilde{\boldsymbol{y}}_{\mathrm{N}}(\boldsymbol{\theta})$.

A usual application is a signal set, where only the upper hemisphere is utilized. A corresponding set of basis functions is shown in fig. 4(c). The truncation must be done carefully as to preserve as much energy as possible in the target surround geometry. Figs. 6 and 5 show the energy for all panning angles when the selected target geometry area is assumed to be hemispherical. To minimize the fluctuation in dB to a value that is smaller than 1dB (just-noticeable difference for stationary broad band noise), fig. 7 shows which

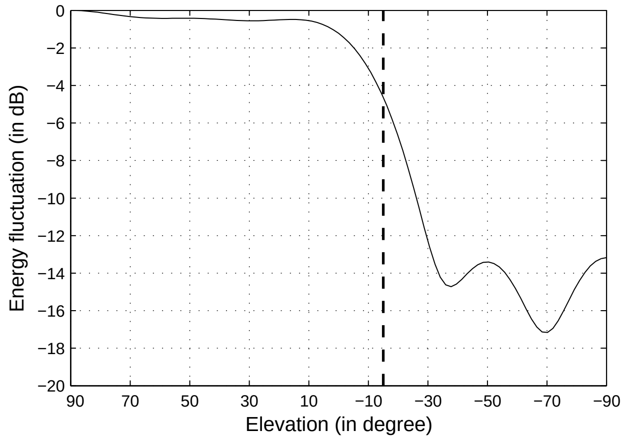


Figure 5: Energy fluctuation of panning on the upper hemisphere, N=4; the dotted vertical line marks the lower integration boundary at -15°.

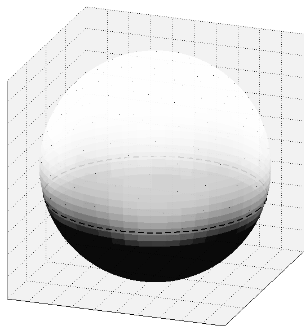


Figure 6: Hemispherically shaped energy fluctuation, N=4; the dotted vertical line marks the lower integration boundary at -15°.

integration interval is suitable: the integration needs to start slightly below the equator to provide an suitable definition and a reduced set of $\mathrm{K} = (\mathrm{N}+1)(\mathrm{N}+1)/2$ basis functions. Eq. (13) in sec. Appendix gives an example of an adaptor matrix for $N = 3$ on the hemisphere.

Moreover, any other spherical fraction can be used as target playback domain, as shown in fig. 2.

5 SOFTWARE/TOOLS

Using Pd as the main composition and musical research environment, IEM has developed a set of tools implementing the ambiX format: the *ambiX euphonic toolbox: AXET*. *AXET* and the full specification will be available at `http://ambisonics.iem.at/ambix`. The toolbox consists of objects for Pd to read/write *basic* and *extended* format and graphical evaluation of the energy distribution. Adaptor matrices for standard conversion scenarios from other formats are available within the toolbox.

This website will be kept up-to-date by the authors and is meant as an open platform for new contributors. The online version of the format enables consideration of thinkable

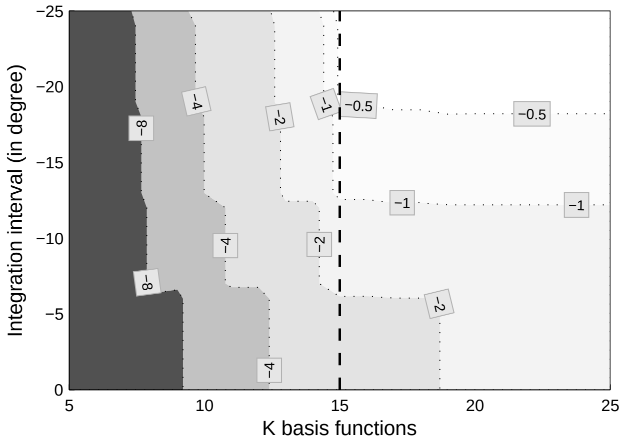


Figure 7: This plot determines the lower integration boundary for basis functions on the hemisphere, N=4. It shows the energy loss [dB] of virtual sources at 90°, N=4. The dotted line shows the expected number of $K = (N+1)(N+2)/2$ basis functions.

improvements and possible new developments, which are hereby strongly encouraged. A database of pre-calculated adaptor matrices for the common use cases (circular, hemispherical) will be provided there.

6 OUTLOOK/CONCLUSION

This paper presented an Ambisonic format that takes into account all thinkable arguments from the practical and academic world using Ambisonics. A long and necessary discussion of available conventions has been given, and finally synthesized into a *basic* and *extended* ambiX format specification. ambiX is future proof as it avoids limitations on the Ambisonic order and file length. The *basic* format fulfills the practical requirement of having a well-defined and unique channel sequence and normalization for any hardware/software multichannel audio processing without meta information. The *extended* format is open to surround reinforcement geometries that are not spherically uniform. As only additional information, it uses an adaptor matrix to restore the default channel set of the *basic* format. Available Ambisonic music can be embedded in *extended* ambiX without re-encoding. The powerfulness of the adaptor matrix is demonstrated by several exemplary applications.

The authors and their discussion partners hope that ambiX brings common unification to old and new Ambisonic music, and they are willing to adapt their software. Ambisonics is a hierarchical audio format for transmission of any content [37]. With its past limitations removed by the above proposal, this can finally become true.

7 ACKNOWLEDGEMENTS

The authors gratefully thank the Ambisonics community and the participants in the various discussions at meetings, such as the Ambisonics Symposium. Fons Adriaensen, Michael Chapman and Jörn Nettingsmeier have supported the authors in the later stages of discussion, which is hereby gratefully acknowledged.

REFERENCES

[1] M. Leese, "File Format for B-Format," checked: *2011-08-01*. [Online]. Available: http://www.ambisonic.net/fileformats.html

[2] D. Malham, "Higher order ambisonic systems," abstracted from 'Space in Music - Music in Space', an Mphil thesis by Dave Malham, submitted to the University of York in April 2003, checked: *2011-09-09*, 2003. [Online]. Available: http://www.york.ac.uk/inst/mustech/3d_audio/higher_order_ambisonics.pdf

[3] T. Musil, M. Noisternig, and R. Höldrich, "A Library for Realtime 3D Binaural Sound Reproduction in Pure Data (Pd)," in *Proc. Int. Conf. on Digital Audio Effects (DAFX-05), Madrid, Spain*, 2005.

[4] F. Adriaensen, "AmbDec 0.4.2 User Manual," checked: *2011-09-09*, 2011. [Online]. Available: http://kokkinizita.linuxaudio.org/linuxaudio/downloads/ambdec-manual.pdf

[5] J. Nettingsmeier, "AMBI@Home The search for extrafrontal intelligence," in *Linux Audio Conference 2008, Cologne*, 2008.

[6] J. Daniel, "Représentation de champs acoustiques, application à la transmission et à la reproduction de scénes sonores complexes dans un contexte multimédia," Ph.D. dissertation, Université Paris 6, 2001.

[7] ——, "Spatial sound encoding including near field effect: Introducing distance coding filters and a viable, new ambisonic format," in *AES 23rd International Conference*, 2003.

[8] ISO/IEC JTC 1/SC 29/WG 11/N6591, "Final Draft Amendment 3: AudioBIFS Extension," 2004.

[9] M. Chapman, W. Ritsch, T. Musil, J. Zmölnig, H. Pomberger, F. Zotter, and A. Sontacchi, "A Standard for Interchange of Ambisonic Signal Sets. Including a file standard with metadata," in *Proc. of the Ambisonics Symposium, Graz, Austria*, June 2009.

[10] Apple Inc, "Apple Core Audio Format Specification 1.0," checked: *2011-09-09*, March 2006. [Online]. Available: http://developer.apple.com/library/mac/#documentation/MusicAudio/Reference/CAFSpec/

[11] The Ambisonics Association, "Ambisonic Channels," checked: *2011-09-09*. [Online]. Available: http://ambisonics.ch/standards/channels/

[12] E. Deleflie, "How to make a universal ambisonic (*.ua) file," checked: *2011-08-01*. [Online]. Available: http://soundofspace.com/static/make_ua_file

[13] C. Travis, "A new mixed-order scheme for ambisonic signals," in *Proc. of the Ambisonics Symposium, Graz, Austria*, June 2009.

[14] E. Merzbacher, *Quantum Mechanics*, 3rd ed. John Wiley & Sons, Inc., 1998.

[15] R. Green, "Spherical Harmonic Lighting: The Gritty Details," in *Game Developer's Conference*, vol. 2003, 2003.

[16] J. Krivanek, J. Konttinen, S. Pattanaik, and K. Bouatouch, "Fast approximation to spherical harmonic rotation," *Publication interne No 1728*, 2005.

[17] R. Pail, G. Plank, and W. Schuh, "Spatially restricted data distributions on the sphere: the method of orthonormalized functions and applications," *Journal of Geodesy*, vol. 75, no. 1, pp. 44–56, 2001.

[18] J. Keiner, S. Kunis, and D. Potts, "Efficient reconstruction of functions on the sphere from scattered data," *Journal of Fourier Analysis and Applications*, vol. 13, no. 4, pp. 435–458, 2007.

[19] A. Kudlicki, M. Rowicka, M. Gilski, and Z. Otwinowski, "An efficient routine for computing symmetric real spherical harmonics for high orders of expansion," *Journal of applied crystallography*, vol. 38, no. 3, pp. 501–504, 2005.

[20] M. Blanco, M. Flórez, and M. Bermejo, "Evaluation of the rotation matrices in the basis of real spherical harmonics," *Journal of Molecular Structure (THEOCHEM)*, vol. 419, no. 1-3, pp. 19–27, 1997.

[21] D. Pinchon and P. Hoggan, "Rotation matrices for real spherical harmonics: general rotations of atomic orbitals in space-fixed axes," *Journal of Physics A: Mathematical and Theoretical*, vol. 40, p. 1597, 2007.

[22] A. Edmonds, *Angular momentum in quantum mechanics*. Princeton University Press, 1957.

[23] "SphericalHarmonics Class (Microsoft.DirectX.Direct3D)," checked: *2011-09-09*. [Online]. Available: http://msdn.microsoft.com/en-us/library/bb296236(v=vs.85).aspx

[24] E. Whittaker and G. Watson, *A course of modern analysis*, 4th ed. Cambridge University Press, 1927.

[25] W. E. Byerly, *An Elementary Treatise on Fourier's Series and Spherical, Cylindrical, and Ellipsoidal Harmonics with Applications to Problems in Mathematical Physics*. The Athenaeum Press, 1893.

[26] E. Heine, *Handbuch der Kugelfunctionen*. Georg Reimer, Berlin, 1861.

[27] M. Ferrers, *An elementary treatise on spherical harmonics and subjects connected with them*. MacMillan and Co., London, 1877.

[28] E. Williams, *Fourier acoustics: sound radiation and nearfield acoustical holography*. Academic Press, 1999.

[29] NIST Digital Library of Mathematical Functions, "Chapter 14 Legendre and Related Functions," checked: *2011-09-09*. [Online]. Available: http://dlmf.nist.gov/14

[30] D. G. Malham and A. Myatt, "3D Sound Spatialization using Ambisonic Techniques," *Computer Music Journal*, vol. 19, no. 4, pp. 58–70, 1995.

[31] M. A. Poletti, "A Unified Theory of Horizontal Holographic Sound Systems," *J. Audio Eng. Soc.*, vol. 48, no. 12, pp. 1155–1182, 2000.

[32] J. Daniel, "Spatial Sound Encoding Including Near Field Effect: Introducing Distance Coding Filters and a Viable, New Ambisonic Format," in *prepr. 16, Proc. 23rd Int. AES Conf.*, Copenhagen, 2003.

[33] D. Bryant, "WavPack. Hybrid Lossless Audio Com-

pression," checked: *2011-09-09*. [Online]. Available: http://www.wavpack.com

[34] H. Pomberger and F. Zotter, "An ambisonics format for flexible playback layouts," in *Proc. of the Ambisonics Symposium, Graz, Austria*, Graz, 2009.

[35] F. Zotter, H. Pomberger, and M. Noisternig, "Energy-preserving ambisonic decoding," *Acta Acustica united with Acustica*, accepted for publication 2011.

[36] N. Sneeuw, "Global spherical harmonic analysis by least-squares and numerical quadrature methods in historical perspective," *Geophysical Journal International*, vol. 118, pp. 707–716, 1994.

[37] M. A. Gerzon, "Hierarchical Transmission System for Multispeaker Stereo," *J. Audio Eng. Soc.*, vol. 40, no. 9, pp. 692–705, 1992.

8 APPENDIX

8.1. Links to development pages of Ambisonics related software

as of 15th August 2011

*1 **SuperCollider** real-time audio synthesis and algorithmic composition
http://supercollider.sourceforge.net

*2 **UGen plugins for SuperCollider** by Joshua Parmenter
http://sc3-plugins.sourceforge.net

*3 **MAX**
http://cycling74.com/products/maxmspjitter

*4 **Externals for MAX** by Philippe Kocher and Jan Schacher
http://www.icst.net/research/downloads/ambisonics-externals-for-maxmsp

*5 **Plogue Bidule** Advanced Modular Audio Software
http://www.plogue.com/?page_id=56

*6 **Ambisonic Bidules** Plogue Bidule plugin suite for Ambisonic processing
http://www.digenis.co.uk/?page_id=59

*7 **amblib** Ambisonic C++ Library
http://sourceforge.net/projects/amblib

*8 **Ardour** Digital Audio Workstation
http://ardour.org

*9 **AMB-plugins** LADSPA plugins by Fons Adriaensen
http://kokkinizita.linuxaudio.org/linuxaudio

*10 **Pure Data (Pd)**
http://puredata.info

*11 **bin_ambi** binaural Pd external
http://iem.at/projekte/dsp/bin_ambi

*12 **iem_ambi** Ambisonic Pd external
http://gem.iem.at/iem_ambi

*13 **CUBEmixer**
http://ambisonics.iem.at/xchange/products/cubemixer

*14 **AmbDec**
http://kokkinizita.linuxaudio.org/linuxaudio

*15 **B-pan & B-dec**
http://www.dmalham.freeserve.co.uk/vst_ambisonics.html

*16 **B2X Plug-In Suite for Mac OS X**
http://www.radio.uqam.ca/ambisonic/b2x.html

8.2. Adaptor matrix example for N=3 hemispherical playback

$$A_{3,\,16x10} = 10^{-3} \begin{bmatrix} -71.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 48.3 & 0 & 0 & 0 & 0 & 0 \\ -30.3 & 21.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 48.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -34.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 15.1 & -26.1 & 0 & 0 & 0 & 0 \\ 5.5 & 31.4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 15.1 & -26.1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -34.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 39.4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -20.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & -13.4 & -29.8 & 0 & 0 & 0 & 0 \\ 9.7 & 17.6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -13.4 & -29.8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -20.7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 39.4 & 0 \end{bmatrix} \quad (13)$$