Toningenieurprojekt

DIY Mixed Order Ambisonics Microphone Array

Michael Romanov

Supervision: Winfried Ritsch

Graz, July 3, 2018

institut für elektronische musik und akustik

Abstract

In this work a DIY-Mixed-Order-Ambisonics microphone is build. Several approaches to equalize spherical microphones are discussed and the easiest way used by multichannel filters to achieve a good sounding result. The idea here is to build a cheap microphone array that can compete with others available on the market. Also it must be easy to build alone without assuming to much theory so everyone can do this at home.

Acknowledgements

I want to thank Dr. Franz Zotter for the fruitful discussions about the theory of spherical microphone arrays and mathematical questions.

Also i would like to thank my colleagues Nils Meyer-Kahlen and Thomas Deppisch for the support during the recording and listening session.

Contents

1 Introduction

In general spherical microphone arrays try to sample a sphere equidistantly [1]. Perceptually it has been shown, that the spatial resolution on the horizontal plane is higher then for elevation [2]. This leads to the idea to build an microphone array that has a higher resolved angular resolution for and less spherical spherical resolution. Combining a circular 2D array with a spherical 3D array is called Mixed Order Ambisonics and is discussed in several publications [3] [4] [5].

Spherical microphone arrays are very expensive. In this work we try to build a microphone array that is affordable. Equalisation of distortions that come up with the Ambisonics transformation can be cancelled by a spherical measurement techniques or analytic filters, that will be discussed in this work. We try to show if it is possible to build an microphone array that has not perfect microphone placement and will equalize manually in a DAW using multichannel filters.

2 Ambisonics Microphone Theory

2.1 Coordinate System

To describe linear movements in space it is common to use cartesian coordinates. To describe circular movements it is common to use spherical coordinates. The transformation from cartesian coordinates to spherical is performed like

$$r = \sqrt{x^2 + y^2 + z^2} \tag{1}$$

$$\varphi = \arctan \frac{y}{x} \tag{2}$$

$$\vartheta = \arccos \frac{z}{\sqrt{x^2 + y^2 + z^2}} = \arccos \frac{z}{r}. \tag{3}$$

The other way round the computation is performed as follows

$$x = r \cos\vartheta \cos\varphi \tag{4}$$

$$y = r \cos\vartheta \sin\varphi \tag{5}$$

$$z = r \sin\vartheta. \tag{6}$$

2.2 Spherical Harmonics

To get an understanding of how Ambisonics works requires familiarity with the spherical harmonic representation of directivity patterns. An standardized representation has been published in the **ambiX** convention [6]. The spherical harmonics are defined as

$$Y_n^m(\varphi, \vartheta) = N_n^{|m|} P_n^{|m|}(\sin\vartheta) \begin{cases} \sin(|m|\varphi), \; for \; m < 0 \\ \cos(|m|\varphi), \; for \; m \geq 0. \end{cases} \tag{7}$$

whereby n denotes the order and m the degree. $P_n^{|m|}$ are the Legendre functions and $N_n^{|m|}$ represents the normalization function and for the trigonometrical functions and for the Legendre functions. For $N_n^{|m|}$ in **ambiX** is defined with the SN3D normalization in

$$N_n^{|m|} = \sqrt{\frac{2-\delta_m}{4\pi}\frac{(n-|m|)!}{(n+|m|)!}}. \tag{8}$$

2.3 Encoding

We can represent a sound source as a directivity pattern composed by an input signal weighed with the sampled spherical harmonics at a specific direction. It is very similar to the beam forming with microphone signals. Overlaying an omnidirectional microphone with a figure-of-eight (dipole) microphone yields cardioid directivity pattern. Same happens in the spherical harmonics domain. Overlay the omnidirectional w-channel with the spherical figure of eight patterns result in a spherical cardioid beam. Increasing the Ambisonics order leads the beam to be more focused like a super cardioid pattern. The focus of the beam increases with the Ambisonics order n. To bring a mono signal $s(t)$in the Ambisonics domain we just calculate the weights of every spherical harmonic $Y_n^m(\varphi,\vartheta)$ up to the chosen Ambisonics order and a given direction (φ, ϑ), multiply the input signal with the weight for every Ambisonics channel

$$\chi_n^m(t) = Y_n^m(\varphi,\vartheta)s(t), \tag{9}$$

and we end up with the Ambisonics signal $\chi_n^m(t)$ with

$$k = (n+1)^2 \tag{10}$$

Ambisonics channels for 3D Ambisonics. For the 2D case there are

$$k = (2n+1) \tag{11}$$

Ambisonics channels.

2.4 Microphone encoding

To transform any set spherical microphone signals into the Ambisonics domain we calculate the weights of the spherical harmonics at microphone positions φ_i, ϑ_i what results in a matrix $\mathbf{y}$. The pseudo-inverse of this matrix maps the microphone signals $\mathbf{s}(t)$ into the Ambisonics domain

$$\gamma(t) = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\,\mathbf{s}(t). \tag{12}$$

As we can see, the matrix needs to be inverted. To calculate a real inverse there is the same amount of capsules as Ambisonics channels needed.

This transformation of the microphone signals leads to distortions of the Ambisonics signals. To avoid this distortion a set of frequency dependant filters $\mathbf{h}(\omega)$ needs to be applied that equalizes the on-axis frequency response and yields undistorted Ambisonics signals

$$\chi(t) = \mathrm{diag}(\mathbf{h}(\omega))\,\gamma(t). \tag{13}$$

2.5 Spatial Filters

This kind of distortions are caused by the geometry of the microphone array, and are basically dependent on the radius of the used microphone array and the amount of capsules used. In general this results in low frequency loss with growing Ambisonics order. On the other hand by growing frequency spatial aliasing comes into play. Radial focusing filters need to be applied to avoid the low frequency loss. These can be either calculated analytically or me measured. Analytical filter design is presented here, here, here and there. In general this introduces a very high gain for low frequencies with growing Ambisonics order. To avoid a very high gain at low frequencies a trade of between spatial resolution in low frequencies and low frequency gain must be taken into account.

Several methods for calculating such filters are discussed in various papers [7] [8] [9] [10].

Measurement of frequency responses can also lead to such filters. A method to do so is discussed in [11].

A comparison of several model based and measurement based approaches to generate encoding filters are discussed in [12]. It shows that measurement and matrix inversion approaches lead to petter performance for low frequencies.

3 Generating Filters

3.1 Theory

Angelo Farina et al. have published a measurement method for synthesizing virtual microphones that can also be used calibrating spherical microphone arrays [11] for Ambisonics. It is generally based on the idea that we try to synthesize virtual microphones that means that a given microphone array is equalized and summed in a way that it represents a microphone with a target directivity pattern. In the presented method the virtual microphone arrays can be any given target pattern. The idea is to measure the directivity pattern of every single microphone of an spherical microphone array from multiple directions and analyse the directivity pattern of every capsule. Given a target directivity pattern a set of filters can be synthesized to create such a pattern by a matrix convolution.

So we take a given set of M real microphones and want to create V virtual microphones with target directivity. For this conversion a set of filters is needed with the dimension MxV. This can be implemented as FIR filters as they are always stable and can be derived from measurements.

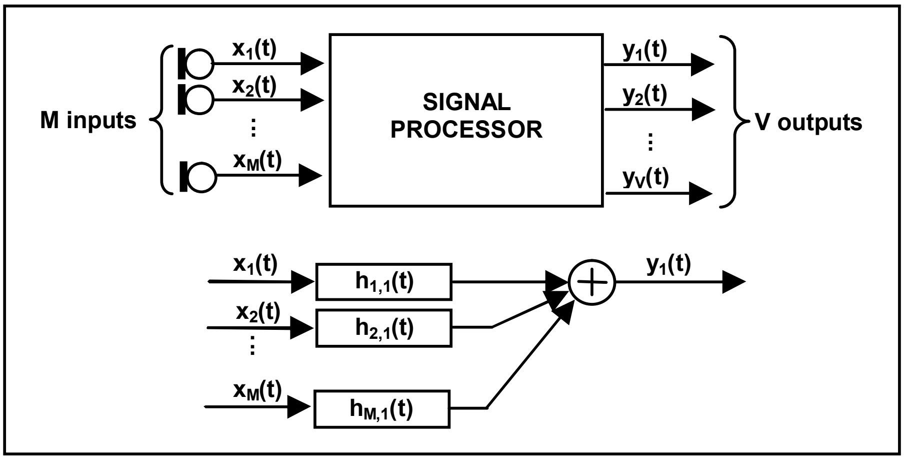

Figure 1 – Rotation computing [?]

So a set of real microphones providing the signals $x_m(t)$ needs to be convolved with with a set of impulse responses $h_{m,v}$ and summed to result a virtual microphone signal $y_v(t)$

$$y_v(t) = \sum_{m=0}^{M} x_m(t) * h_{m,v}. \tag{14}$$

In Farinas approach the virtual microphones are final directivity patterns avoiding Ambisonics decoding and encoding. This principle allows synthesizing virtual microphones having an arbitrary directivity patterns.

For this work we will use the more basic approach what is also discussed in Farinas paper. Instead of synthesizing a complex given microphone patterns our target directivity functions are the spherical harmonics. This allows us to create a Ambisonics microphone that outputs normalized and equalized Ambisonics signals.

So the filters we want to derive are the spacial equalizing filters that equalize the on axis impulse responses of the spherical harmonics. This way we end up with virtual microphones that represent the spherical harmonics and we can apply all kind of Ambisonics effects and manipulations to it.

A target spherical microphone directivity pattern $Q_n(\varphi, \vartheta)$ is assumed, in our case this represents the real valued spherical harmonic functions.

The characterization of the spherical microphone array can be described as a matrix of measured anechoic impulse responses C generated with a on this distance equalized sound source placed at a large number D of positions all around the probe.

So matrix C has the dimensions $M \times D$. To derive such set of filters $\mathbf{h}$ we should transform the measured impulse responses c into the theoretical impulse responses p:

These theoretical responses p are derived from the target directivity pattern $Q_n(\varphi, \vartheta)$ applied to a delayed unit-amplitude Dirac's delta function δ. So for one specific direction d out of D directions in total this results in

$$p_d = \sum_{m=0}^{M} c_{m,d} * h_m. \tag{15}$$

An easier way to compute such filters is in the frequency domain by computing the complex spectra using FFT algorithm to the N-points-long impulse responses c, h and p this results in

$$P_d = \sum_{m=0}^{M} C_{m,d,k} \cdot H_{m,k} \tag{16}$$

for each frequency bin k whereby P, C and H denote the resulting complex spectra.

Now we can calculate a matrix of filters $\mathbf{H}$ for the V virtual microphones (here spherical harmonics) by

This yields an over-determined system and doesn't admit an exact solution. However, it is possible to approximate a solution using the Least Squares method for matrix inversion employing a regularization technique to avoid instabilities and excessive signal boost.

To calculate the Least Squares solution following system is applied.

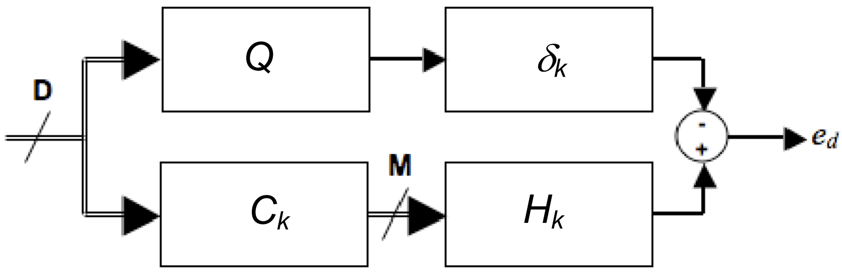

Figure 2 – Rotation computing [**?**]

A delay δ is introduced here. This one is needed to produce causal filters. We can also observe a modelling error e that is needed by the Least Squares algorithm to be minimized.

In general in frequency-domain the delay function is represented as

$$\delta_k = e^{-j2\pi k \frac{n_0}{N}} \tag{17}$$

The easiest way to apply a suitable delay is setting n_0 to $N/2$. Hereby the peak of the impulse response moves to the middle and the decay moves to the beginning and the end of the impulse response.

As we want to avoid excessive emphasis at frequencies where the signal is very low a regularisation parameter β is needed in the dominator of the matrix calculation.

Using the psoydo inverse and target pattern yields the searched filter

$$\mathbf{H} = \mathbf{C}^T(\mathbf{C}\mathbf{C}^T + \beta(\omega)\mathbf{I})^{-1} Q e^{-i\omega\tau} \tag{18}$$

The regularization parameter β depends on frequency ω and should be set up properly according to the frequency range where the microphone array is designed to work properly. In this frequency range it needs to be specified as a very small value. For frequencies where conditioning problems could cause numerical instability of the solution we use larger values e.g. very low and very high frequencies with a smooth transition between the central band and the outer bands.

A a good choice for the spectral shape of the regularization parameter is to specify it as a small, constant value inside the frequency range where the probe is designed to work optimally, and as much larger values at very low and very high frequencies, where conditioning problems are prone to cause numerical instability of the solution.

In general this is all what is needed to calculate such filters and is implemented in python as a script that inputs impulse responses and results in the needed filter impulse responses to equalize the microphone array and is part of the AAMA project (`https://git.iem.at/cm/AAMA`)[13].

3.2 Model

Another approach to generate such filters is having a mathematical model of the microphone. In general just the radius and the order of the microphone is needed. So assuming shadowing effects the theoretical distortion of the microphone array can be calculated. Assuming equal frequency response of the microphones and using only one measurement from one direction is sufficient enough to fit all the curves of the model into this measurement. So the needed filters can be easily calculated.

4 Microphone Construction

We use 16 Behringer ECM8000 measurement microphone as they appear to have a pretty flat frequency response and are pretty small standard deviation in frequency response.

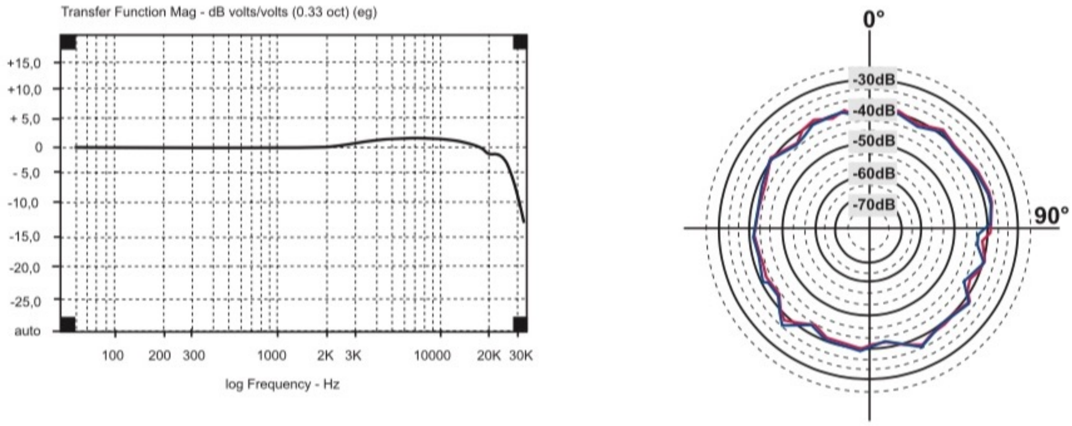

Figure 3 – Frequency Response Behringer ECM8000

4.1 De-assembling Behringer Measurement MIc

To build this microphone we use 16 Behringer ECM8000 measurement microphones. These are discussed in several forums as a wonder microphone.

To de-assemble these microphones we use a heat gun to get rid of the glue that holds the cap with the capsule inside as shown in Fig. 4.

Figure 4 – Heat gun

After heating we carefully use a pincer and some paper to not scratch the material to get off the capsule as can be seen in Fig. 5.

Figure 5 – Pincer

In a last step we take out the screws and can take the PCB out of the aluminium tube. The inner life of the microphone is shown in Fig. 6. It is needed to transform the $48V$ symmetrical phantom power to the 3V the electret capsules need.

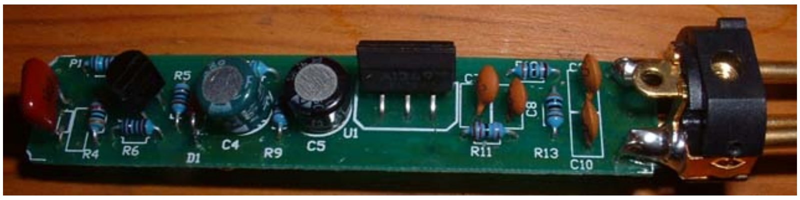

Figure 6 – PCB

4.2 Sphere Design

A publication on psychoacoustics show that the angular resolution of spatial hearing is much higher for the horizontal plane as for the perception of height [2]. So the idea here is to use a 4^{th}-order circular 2D microphone array for azimuth, whereby a higher order can be achieved by less amount of capsules. For the 3D part we use a 2^{nd}-order 3D microphone array. The coordinates used are shown in Tab. 1

Table 1 – Microphone Layout

lspk.idx	elevation / °	azimuth / °
1	0	0
2	40	0
3	80	0
4	120	0
5	160	0
6	-160	30
7	-120	30
8	-80	30
9	-40	30
3	0	90
4	0	45
5	120	45
6	-120	45
7	60	-45
8	180	-45
9	-60	-45

4.3 Building the microphone

This microphone array is build of "FIMO Light air" a modelling material that dries on air or in the microwave. We use a styrofoam bowl to form the microphone and arrange 9 microphones on a circle and 6 in triangles.

The Microphone has an radius of $5cm$.

Figure 7 – Sphere formed out of FIMO air light

The capsules are connected to the PCB using ribbon cable. The 16 PCBs output a ribbon cable with 48 wires that will afterwards be connected via XLR to the audio interface.

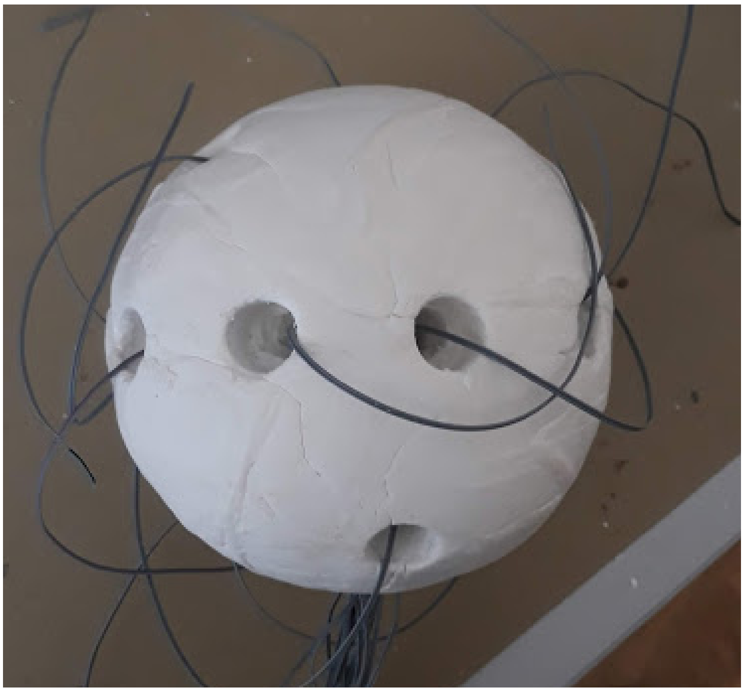

Figure 8 – Box holding 16 PCBs

All the 16 PCBs were packed into a box that can be placed nearby the microphone. The output of the box.

Figure 9 – Box holding 16 PCBs

5 Measurement

5.1 Measurement Setup

To measure the impulse responses c for D directions we install the spherical microphone array on a turntable that can be turned in predefined steps that define our measurement directions.

The measurements will be performed in an almost anechoic chamber at the IEM Petersgasse 116.

5.2 Calibration

We place the loudspeaker $1m$ away from the microphone array. We need to equalize the "air-channel" between the loudspeaker and the microphone first to be sure that the exponential sine sweep that we use as measurement signal has an equalized frequency response. Therefore we place an calibrated measurement microphone on the position of the sphere and do a measurement. Deconvolving the "air-channel" with the measuerement signal in frequency domain

$$c_{correct}(t) = IFFT(\frac{FFT(c_{air}(t))}{FFT(c_{sweep}(t))}) \tag{19}$$

yields the filter to be applied to our impulse response to get an equal measurement signal for our measurement on the turntable.

The microphone was positioned in an anechoic chamber on an turntable that can be rotated in $0.5°$ steps.

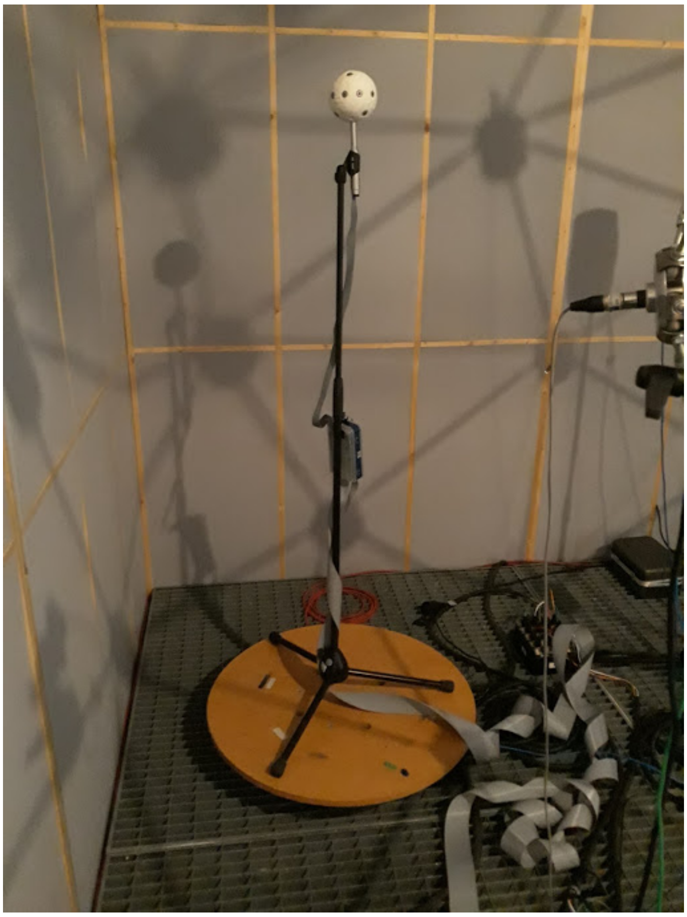

Figure 10 – Controllable turntable

It can be controlled over ethernet. So using a software e.g. Pure Data can be used to automate the measurement. The software plays a sweep and records the responses to the sweep on all 16 capsules.

For the measurements an Genelec 8030 Loudspeaker was positioned $1m$ away from the microphone array. A reference measurement was recorded using one measurement microphone at the position of the array to equalize the channel between measurement microphone and the loudspeaker as Fig. 11 shows.

Figure 11 – Box holding 16 PCBs

4 measurement series with 2 repetitions have been performed. First one the microphone in upward position (Fig. 13) and rotated in 10° steps the microphone number one facing the loudspeaker in the beginning of the procedure.

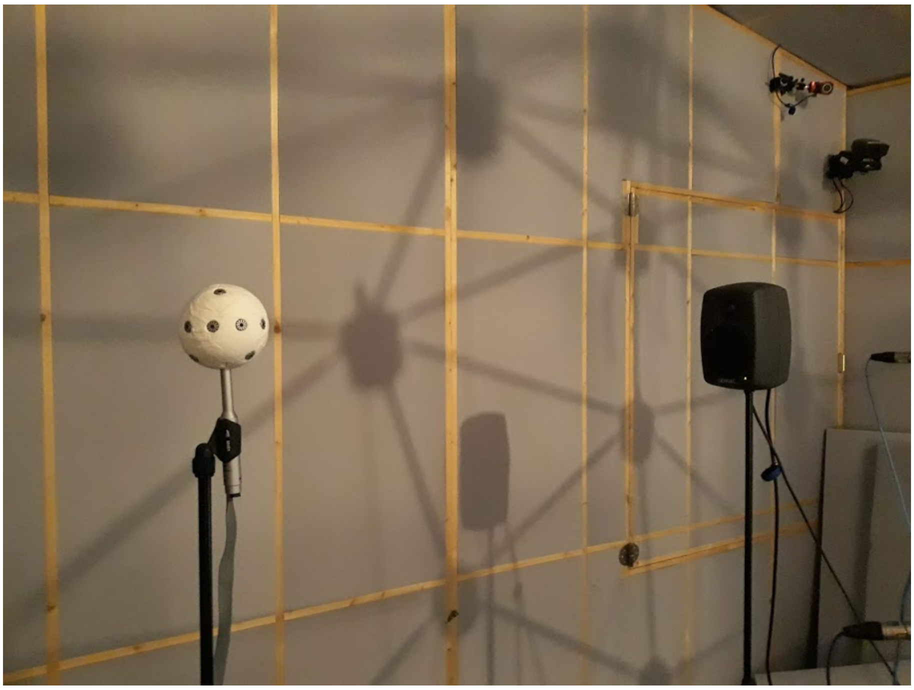

Figure 12 – Azimuth Measurements

Second measurement was performed with the first microphone facing the floor and the top microphone facing the loudspeaker.

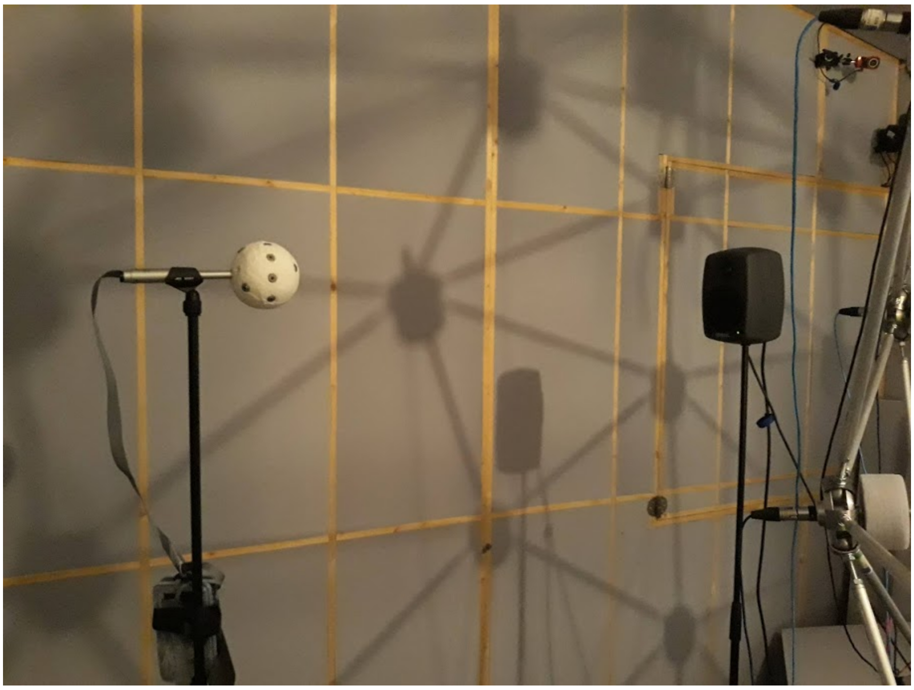

Figure 13 – Azimuth Measurements

For measurement three the microphone array was turned around for 90° and 225° for th fourth one. When looking towards the loudspeaker the rotation was performed clockwise.

The full process results in $36 \times 4 = 144$ measurement positions that can be used for filter generation and equalizing.

6 Encoding and Filtering

6.1 Encoding

For the encoding of the microphone signals we use the MultiEncoder Plug-in from the IEM Plugin Suite [14]. For the first 9 channels we position the sources according on the horizontal plane using 4^{th}-order, what results in only bringing signals to the circular harmonics (see Fig. 14).

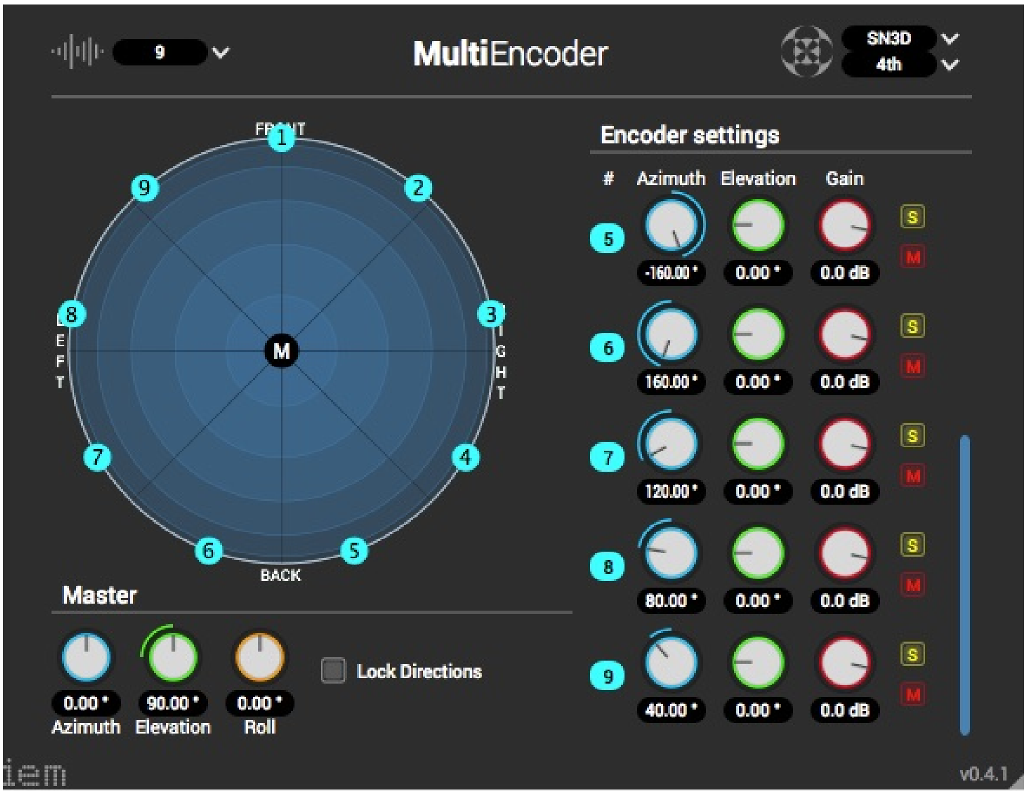

Figure 14 – Encoding the 2D ring microphone signals

The rest of the microphone signals are encoded 2^{th}-order what can be observed in Fig. 15.

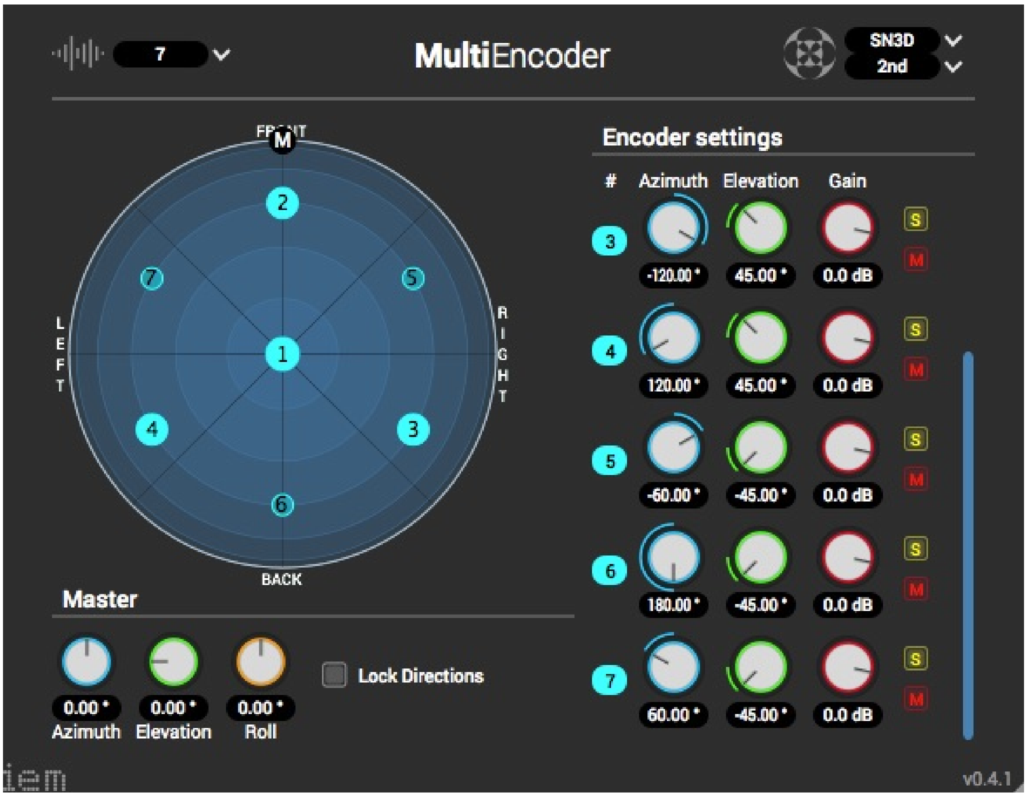

Figure 15 – Encoding the 3D spherical microphone signals

Afterwards the signals are summed up on the Ambisonics bus for further equalisation.

6.2 Filtering

A set of multichannel filters are used to approximate the needed filters. For HOA microphone arrays a tool was published in [15], but the current version does not support MOA arrays.

The target filters to be applied to a microphone with $10cm$ radius can be seen in Fig. 16

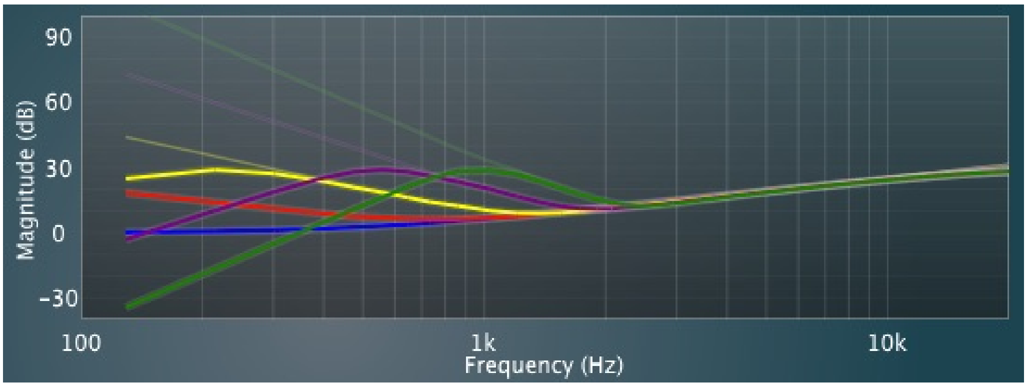

Figure 16 – Sparta - array2sh [15]

Filters for the given microphone array. Thin lines represent the real needed filters with very high gain in the low frequencies. The thick lines show the Thikonov filters that we will approximate to boost low frequencies with maximum $30dB$ in the needed frequency range. Hereby the blue lines represents the 0^{th}-order filter, the red lines represent the 1^{th}-order filter, yellow line the 2^{th}-order filter, purple line the 3^{th}-order filter and the green one represents the 4^{th}-order filter.

As all orders have the same high shelf characteristic we put a mcfx-filter [16] with high shelf characteristic on the sum bus of the encoded signals as can be seen in Fig. 17.

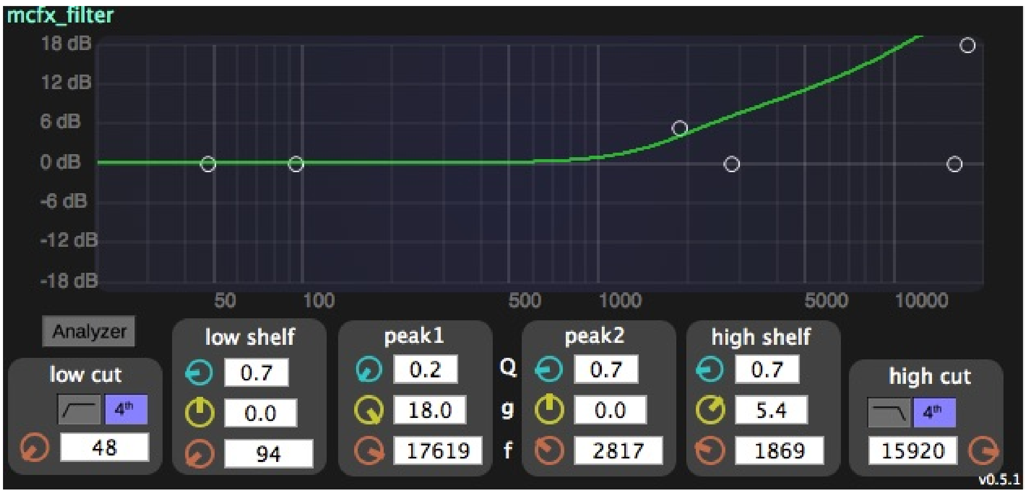

Figure 17 – mcfx-filter [16]

For the other channels we use a low shelf and a peak filter for the lower frequencies as shown exemplary for the 3^{th}-order channels in Fig. 18.

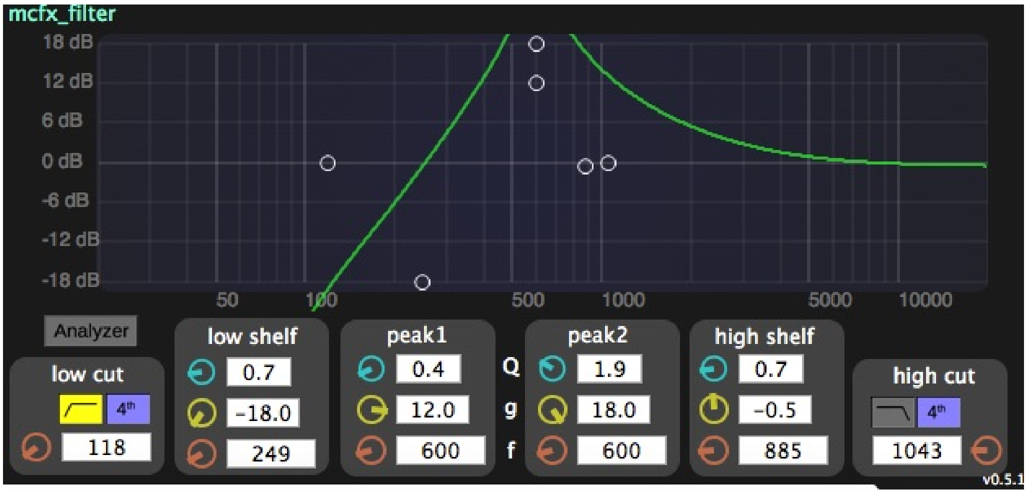

Figure 18 – mcfx-filter [16]

For the other Ambisonics order channels we behave similarly at the needed frequencies. With reaper Plug-in pin connectors we can decide what channels to apply for the filtering.

7 Microphone Comparison

7.1 Recording Session

To compare the built microphone to some other available microphone arrays as set of test recordings was created. Therefore we positioned 6 microphones (DIY-Mic, Eigenmike, Oktava, Tetramic, CoreSound and Zoom H2N) next to each other in a seminar room what can be seen in Fig. 19.

Figure 19 – Microphone comparison

A few scenes were recorded. One type was walking around in a circular trajectory with a shaker. Second scene were positioning the shaker one meter away from the array in front, left, right and back position for $0°$ and $45°$ elevation. Also for $90°$ elevation.

Another scene were three people talking, first speaker talk after each other and afterwards all together. With this scene it can be tested how good virtual microphones work for any array.

The positions we said before any recording.

Last set of recordings was done using a guitar making a figure of eight walktrough the room. Using this scene it can be compared how natural the guitar sounds and also how the impression of closeness changes during the movement.

7.2 Listening Session

In a quick listening session the microphones were compared. The DIY-microphone array sounds very similar to the Eigenmike in terms of sound colour, but introducing more noise compared to the Eigenmike. Also more environmental noise such as the air conditioner can be heard on the DIY microphone, this caused by not using a microphone suspension.

As expected we hear a better spatial resolution of the DIY-microphone that to the 1^{st}-order microphones but less than the Eigenmike. The DIY-microphone sounds more natural than the Eigenmike.

In general it is a good sounding result of this simple approach. The microphone sounds suitable and the easy encoding en filtering leads to a sufficient listening experience. However, the comparison between several filter approaches was not performed here due to a lack of time.

However this was just a short listening session. A proper perceptive evaluation needs to be done to scientifically prove this comparison using the recorded material. This could be done similar to [17] [18] and [19].

8 Conclusion and Outlook

An Mixed-Order-Microphone array was build to prove if it is possible to achieve such an microphone with less then 500 Euro costs. Several techniques for equalisation of spherical harmonics channels were discussed. A set of measurements (144 directions) were made for further evaluation. To equalize the microphone a set of multichannel filters were used in a DAW that achieved a suitable result.

In general it is cheap and quite easy to build spherical microphone arrays. But at the end an multi channel audio interface, pre-amps and a computer is needed to make recordings. This makes field recordings pretty complicated. In a further work an piece of hardware can be designed to make a 16-channel field recorder. An approach can be to use the dsPIC33 Digital Signal Controllers. They combine DSP capabilities with micro-controllers and can process 16 channels of 16 bit audio in 44100 Hz sampling rate on one chip and can be easily programmed in C. In combination with four Analog Devices 4 channel - preams/adc an field recorders can be designed that includes equalization of the microphones due convolving and directly writing to an SD-card. This way a very handy tool can be created in a further work.

Also a set of recordings were made to compare 6 different spatial audio microphones.

As a next step it would be interesting to make an perceptual evaluation using the material recorded during this work in regard to sound colouration, source extent, depth graduation. It can also be evaluated how the several equalizing approaches effect the quality of the microphone array.

References

[1] F. Zotter, *Analysis and Synthesis of Sound-Radiation with Spherical Arrays*. PhD thesis, December 2009.

[2] R. David, D.R. Perrott and K. Saberi, "Minimum audible angle thresholds for sources varying in both elevation and azimuth," 1990.

[3] S. Favrot,; M. Marschall, "Metrics for performance assessment of mixed-order Ambisonics spherical microphone arrays," 2012.

[4] M. Marschall, S. Favrot, and J. Buchhol, "Robustness of a mixed-order Ambisonics microphone array for soundfield reproduction," April 2012.

[5] J. Trevino, S. Koyama, S Sakamoto and Yoiti Suzuki, "Mixed-order Ambisonics encoding of cylindrical microphone array signals," 2014.

[6] C. Nachbar, F. Zotter, E. Deleflie, A. Sontacchi, "AMBIX - A Suggested Ambisonics Format," June 2011.

[7] S. Lösler, F. Zotter, "Comprehensive Radial Filter Design for Practical higher-order Ambisonic Recording," 2015.

[8] F. Zotter, M. Frank, C. Haar, "Spherical microphone array equalization for Ambisonics," 2015.

[9] R. Baumgartner, H. Pomberger and M. Frank, "Practical Implementation of Radial Filters for Ambisonic Recordings," 2015.

[10] F. Zotter, "A Linear-Phase Filter-Bank Approach to Process Rigid Spherical Microphone Array Recordings," June 2018.

[11] A. Farina, A. Capra, L. Chiesi, L. Scopece, "A Spherical Microphone Array For Synthesizing Virtual Directive Microphones In Live Broadcasting And In Post Production," October 2010.

[12] A. Politis, H. Gamper, "Comparing Modeled and Measurement-Based Spherical Harmonic Eencoding Filters For Spherical Microphone Arrays," October 2017.

[13] W. Ritsch, "Affordable Ambisonics Microphones Array - DIY higher order ambisonics microphone kits - https://git.iem.at/cm/AAMA," 2017.

[14] D. Rudrich, "IEM Plug-in Suite - https://plugins.iem.at," 2018.

[15] L. McCormack, S. Delikaris-Manias, A. Farina, D. Pinardi, V.Pulkki, "Real-Time Conversion of Sensor Array Signals into Spherical Harmonic Signals with Applications to Spatially Localized Sub-Band Sound-Field Analysis ," May 2018.

[16] M. Kronlachner, "mcfx - multichannel audio plugin-suite http://www.matthiaskronlachner.com/?p=1910," 2018.

[17] E. Bates, S.Dooney, M. Gorzel, H. O'Dwyer, L. Ferguson and F. M. Boland, "Comparing Ambisonic Microphones Part 1," July 2016.

[18] E. Bates, S.Dooney, M. Gorzel, H. O'Dwyer, L. Ferguson and F. M. Boland, "Comparing Ambisonic Microphones Part 2," May 2017.

[19] E. Kurz, F. Pfahler, M. Frank, "Comparison of first-order Ambisonics microphone arrays," September 2015.