

Audio Engineering Project

Plug-In for Frequency-Dependent Control of Microphone Polar Patterns

Thomas Deppisch

Supervisors: Dr. Matthias Frank, Dr. Franz Zotter

Graz, July 7, 2020



institute of electronic music and acoustics



Abstract

Arbitrary first-order polar patterns can be created by weighted summation of the outputs of front and back diaphragm of a dual-diaphragm microphone. If the output signals of both diaphragms are accessible inside a digital audio workstation (DAW), the polar pattern can be (re)adjusted at any time and is not limited to discrete, pre-defined patterns. This work describes the development of an open-source software plug-in, which allows for the frequency-dependent control of a dual-diaphragm microphone's polar pattern in up to five adjustable frequency bands. Two approaches for creating the underlying filterbank are presented: A linear-phase FIR approach and an IIR approach. Diffuse-field and free-field equalization of synthesized polar pattern signals are described exemplarily for a particular microphone and a filter for compensation of the proximity effect is derived from a physical model. An algorithm for automatic polar pattern optimization is designed, based on the intensity of a target signal.

Zusammenfassung

Durch gewichtete Summierung der beiden Ausgänge eines Doppelmembranmikrofons können beliebige Richtcharakteristiken erster Ordnung erstellt werden. Wenn die Ausgänge beider Membranen in einer Digital Audio Workstation (DAW) verfügbar sind, kann die Richtcharakteristik jederzeit angepasst werden und ist nicht auf diskrete, vordefinierte Richtcharakteristiken begrenzt. Diese Arbeit beschreibt die Entwicklung eines quelloffenen Plug-Ins zur Steuerung der Richtcharakteristik eines Doppelmembranmikrofons in bis zu fünf Frequenzbändern. Es werden zwei Ansätze zur Erstellung der Filterbank vorgestellt: Ein linear-phasiger FIR-Ansatz und ein IIR-Ansatz. Am Beispiel eines bestimmten Mikrofons werden Diffusfeld- und Freifeldverzerrung der synthetisierten Richtwirkung beschrieben. Ein Filter zur Kompensation des Nahbesprechungseffekts wird von einem physikalischen Modell abgeleitet. Zur automatischen Optimierung der Richtcharakteristik wird ein Algorithmus basierend auf der Intensität eines Zielsignals entworfen.

Contents

1	Introduction	5
1.1	First-order polar patterns	5
2	Filter banks: Parametric filter design	7
2.1	Frequency transformations	8
2.2	IIR approach	8
2.3	FIR approach	10
3	Equalization filters: Measurement-based filter design	12
3.1	Motivation	12
3.2	Free-field equalization	13
3.3	Diffuse-field equalization	14
3.4	1/n-octave band smoothing	16
3.5	Minimum-phase filters via cepstral domain	16
3.6	Magnitude responses of the final filters	17
4	Proximity compensation: Model-based filter design	20
4.1	The proximity effect	20
4.2	Continuous-time compensation filter	20
4.3	Discretization: Comparison of impulse invariance and bilinear transform .	21
4.4	Normalization and comparison to measurements	23
5	Automatic polar pattern optimization	25
6	Implementation	26
7	Conclusion and Outlook	28

1 Introduction

Conventional single-output, dual-diaphragm microphones enable the user to set the broadband polar pattern to one of a few predefined polar patterns. When changing the polar pattern, the pole voltages of both diaphragms are adjusted internally, such that the summed signal of both diaphragms corresponds to the chosen polar pattern. Recently, some microphone manufacturers started building dual-diaphragm microphones, where the outputs of both diaphragms can be accessed individually. Such microphones are referred to as dual-output microphones in this work. Dual-output microphones give the recording and mixing engineer more freedom of choosing a custom polar pattern in post production, if corresponding software tools are available. In this project thesis, such a software tool is developed as open-source plug-in, allowing for frequency-dependent control of polar patterns in up to five frequency bands.

Special consideration is given to filter design, which is approached in three different ways: parametric filter design in case of a filter bank (section 2), measurement-based filter design for free-field and diffuse-field equalization (section 3), and model-based filter design in case of a compensation filter for the proximity effect (section 4). As these are very common filter design tasks, this work can be used as a guide, showing different ways of how to approach filtering problems in audio signal processing.

Additional to the different filter design tasks, section 5 describes an algorithm for automatic optimization of the polar pattern based on a target signal. Section 6 gives an overview of the plug-in implementation.

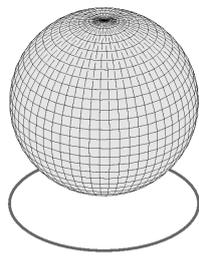
This work was carried out in cooperation with the Austrian Audio GmbH. Measurement-based filters are designed specifically for the Austrian Audio OC818 microphone.

1.1 First-order polar patterns

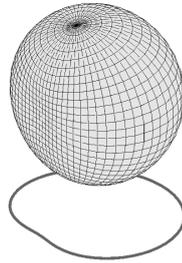
First-order polar patterns s can be created by weighted summation of an omnidirectional component s_0 and a figure-of-eight component s_8

$$s = (1 - \alpha)s_0 + \alpha s_8, \quad (1)$$

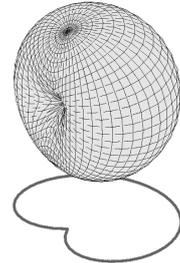
where the weight $\alpha \in [0, 1]$ defines the resulting pattern. In principle, instead of omnidirectional and figure-of-eight, any two different coincident first-order patterns operating on the same axis can be superimposed, allowing the creation of arbitrary first-order polar patterns, if the correct weights are applied. In case of two opposing cardioids, as often seen in dual-diaphragm microphones, the corresponding omnidirectional and figure-of-eight characteristic can be created by summation and subtraction of the two cardioids, respectively. Some commonly used first-order polar patterns and their weighting factors α are depicted in figure 1.



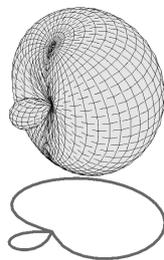
(a) omnidirectional, $\alpha = 0$



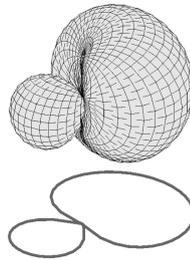
(b) sub-cardioid, $\alpha = 0.37$



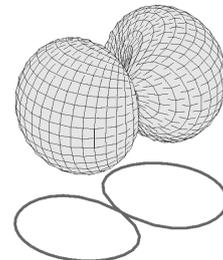
(c) cardioid, $\alpha = 0.5$



(d) super-cardioid, $\alpha = 0.634$



(e) hyper-cardioid, $\alpha = 0.75$



(f) figure-of-eight, $\alpha = 1$

Figure 1 – 3D shapes and horizontal cuts through first-order polar patterns with different weighting factors α .

2 Filter banks: Parametric filter design

Filter banks split signals into subband signals by attenuation of signal content outside the respective subband. This process is accomplished by a parallel structure of a lowpass filter, several bandpass filters and a highpass filter, whose impulse responses can either be of infinite length (IIR) or of finite length (FIR). These well-known filter types are characterized by parameters such as steepness, stopband attenuation, passband and stopband ripple, and group delay. To reduce the complexity of the design process to a single lowpass filter design, the employed filters are often not designed separately, but created from a prototype lowpass filter via frequency transformation. Equivalently, instead of transforming the lowpass filter, the input signal can be shifted to the baseband by complex demodulation, lowpass filtered and shifted back to the original band. In the latter case, the baseband signals can be processed at a lower sampling rate, allowing a computationally more efficient implementation. A second filter per band then needs to band limit the baseband signal before upsampling to avoid aliasing. Typically, demodulation to the baseband, lowpass filtering and downsampling are combined in the analysis stage, while the second filtering step, upsampling and modulation to the original frequency band are combined in the synthesis stage.

Common implementations of filter banks include Polyphase Networks (PPN) and Quadrature Mirror filter banks (QMF), an overview is given in [LV08]. These kinds of filter banks can be designed i) very efficiently by using a multirate structure involving down- and up-sampling, and ii) to be perfectly reconstructing, meaning that the summed output of the filter bank $y[n]$ is only a delayed, scaled version of the input $x[n]$

$$y[n] = cx[n - n_0], c \in \mathbb{R}_{\neq 0}. \quad (2)$$

In case of PPNs, the discrete Fourier transform (DFT) is used as frequency transformation to generate narrow band filters and in case of QMFs a frequency transformation creates highpass filters to recurrently divide the spectrum into half-bands. Non-uniform filter banks can be created with QMFs by splitting one or more half-bands more often than other half-bands. Still, each band can only be divided into two half-bands of equal bandwidth, so arbitrary, non-uniform bandwidths cannot be accomplished and hence, QMFs are not considered in the following. In case of PPNs, non-uniform bandwidths can be accomplished via frequency warping [LV05].

In section 2.1 common frequency transformations are discussed as they are utilized to create subband filters from a prototype lowpass in many filter bank implementations. Section 2.2 and 2.3 investigate two approaches for creating the five-way filter bank: An IIR approach using cascaded Linkwitz-Riley filters with the advantages of computational efficiency and no introduction of a general delay, and an FIR approach with the advantage of a linear phase response which implies constant group delay. These examples show the underlying principle of two filter banks and how they can be implemented with lowest complexity. Still, both approaches can be integrated into more complex multirate or polyphase structures if computational efficiency needs improvement.

2.1 Frequency transformations

Frequency transformations [Bro66][Con70], also called frequency warping [LV08] or (complex) modulation [LV05], allow shifting and in some cases stretching of frequency responses of digital signals or filters. As this work is targeted at a particular software application, only discrete-time z-domain transformations are considered. Nevertheless, similar transformations are applicable to continuous-time filtering problems. Shifting a filter's frequency response by ω_0 in frequency domain $X(\omega - \omega_0)$, can be expressed as a convolution with a dirac impulse at the desired new center frequency $\delta(\omega - \omega_0)$, or as a rotation of ω_0 of the unit circle in z-domain. In time domain, this corresponds to multiplication by a complex exponential function $e^{jn\omega_0}$ or by a cosine function $\cos(n\omega_0)$, to create a symmetric frequency response

$$\mathcal{Z} \{ e^{jn\omega_0} x[n] \} = X(e^{-j\omega_0} z), \quad (3)$$

$$\mathcal{Z} \{ 2 \cos(n\omega_0) x[n] \} = X(e^{-j\omega_0} z) + X(e^{j\omega_0} z), \quad (4)$$

where n denotes the discrete time index. While eq. 4 is used to create a bandpass filter from the prototype lowpass, this equation would double the resulting amplitude for lowpass to highpass transformations with $\omega_0 = \pi$. So, for highpass transformation either eq. 3 must be utilized or an additional scaling factor of $1/2$ is needed

$$\mathcal{Z} \{ \cos(n\pi) x[n] \} = \frac{1}{2} (X(e^{-j\pi} z) + X(e^{j\pi} z)) = X(-z). \quad (5)$$

Constantinides [Con70] derived allpass-based transformations for lowpass filters which additionally include modification of the prototype lowpass' cut-off frequency, which results in frequency warping. These transformations replace the complex z-domain variable z^{-1} of a lowpass filter by an allpass filter $H_A(z^{-1})$

$$z^{-1} \rightarrow H_A(z^{-1}), \quad (6)$$

defining a mapping from the z-plane to another z-plane. The transformation equations for transformation from lowpass to lowpass, highpass, bandpass and bandstop are given in table 1. Löllmann included these transformations into the framework of a PPN, creating an efficient implementation of a non-uniform filter bank, called Filter Bank Equalizer [LV05].

2.2 IIR approach

Conventional parametric design of digital IIR filters is based on analog filter designs and discretization thereof. The underlying analog filter equations stem from approximation of a frequency response via polynomials, such as Butterworth or Chebychev polynomials. To create discrete-time filter equations from these continuous-time designs, Impulse Invariance or Bilinear Transform [OS10, chapter 7.2] are the most often utilized transforms. Section 4.3 includes a comparison of both discretization concepts in context of model-based filter design. For parametric filter design, analytic discrete-time filter implementations are available in signal processing software for most common filter types.

target filter type	$H_A(z^{-1})$	design formulas
lowpass	$\frac{z^{-1} - \beta}{1 - \beta z^{-1}}$	$\beta = \frac{\sin(\frac{\omega_c - \omega_0}{2})}{\sin(\frac{\omega_c + \omega_0}{2})}$
highpass	$\frac{-z^{-1} - \beta}{1 + \beta z^{-1}}$	$\beta = -\frac{\cos(\frac{\omega_c + \omega_0}{2})}{\cos(\frac{\omega_c - \omega_0}{2})}$
bandpass	$-\frac{z^{-2} - \frac{2\beta\kappa}{\kappa+1}z^{-1} + \frac{\kappa-1}{\kappa+1}}{\frac{\kappa-1}{\kappa+1}z^{-2} - \frac{2\beta\kappa}{\kappa+1}z^{-1} + 1}$	$\beta = \frac{\cos(\frac{\omega_2 + \omega_1}{2})}{\cos(\frac{\omega_2 - \omega_1}{2})}$ $\kappa = \cot(\frac{\omega_2 - \omega_1}{2}) \tan(\frac{\omega_c}{2})$
bandstop	$\frac{z^{-2} - \frac{2\beta}{1+\kappa}z^{-1} + \frac{1-\kappa}{1+\kappa}}{\frac{1-\kappa}{1+\kappa}z^{-2} - \frac{2\beta}{1+\kappa}z^{-1} + 1}$	$\beta = \frac{\cos(\frac{\omega_2 + \omega_1}{2})}{\cos(\frac{\omega_2 - \omega_1}{2})}$ $\kappa = \cot(\frac{\omega_2 - \omega_1}{2}) \tan(\frac{\omega_c}{2})$

Table 1 – Equations for transformation of lowpass filter with cut-off frequency ω_c to lowpass, highpass, bandpass and bandstop filter [Con70].

The most simple filter bank consists of only two channels and can be realized using a single lowpass-highpass crossover pair. Linkwitz-Riley crossover filters [Lin76], built from two butterworth filters in series, achieve an allpass frequency response, meaning that both outputs sum up to unit-magnitude. D’Appolito [D’A87] showed that a multi-way crossover filter bank using cascaded two-way crossover filters and phase-compensating allpass sections exhibits allpass characteristic, if all the underlying two-way crossovers show allpass characteristics. Further, the group delay of the summed filter bank is equal to the sum of the group delay of the two-way crossover pairs. Such a multi-way filter bank can be created by successively splitting the signal into branches with lowpass and highpass Linkwitz-Riley crossover pairs. For each of these filter pairs $H_{lp,i}$, $H_{hp,i}$, an allpass section ϕ_i compensating for the phase response is needed in all other branches. This approach yields a filter bank achieving the same phase response in every branch if both filters in every two way-pair have an equal phase response. A five-way filter bank built via this approach consists of four two-way crossover pairs and five allpass filter stages and is depicted in figure 2. An allpass filter ϕ_i to compensate for the phase response of a particular two-way crossover pair $H_{lp,i}$, $H_{hp,i}$ can be created by summing up the transfer functions of both two-way filters, which corresponds to connecting them in parallel and summing up the outputs

$$\phi_i = H_{lp,i} + H_{hp,i} . \quad (7)$$

Figure 3 shows the magnitude response and group delay of a five-way filter bank built via this approach. All the two-way highpass and lowpass pairs consist of 4th-order Linkwitz-Riley filters. Highpass filters can be created from the prototype lowpass via the frequency transformation in table 1. As the group delay is calculated as the sum of the two-way pairs’ group delay, higher filter orders, a higher number of branches and lower crossover frequencies usually lead to larger group delays.

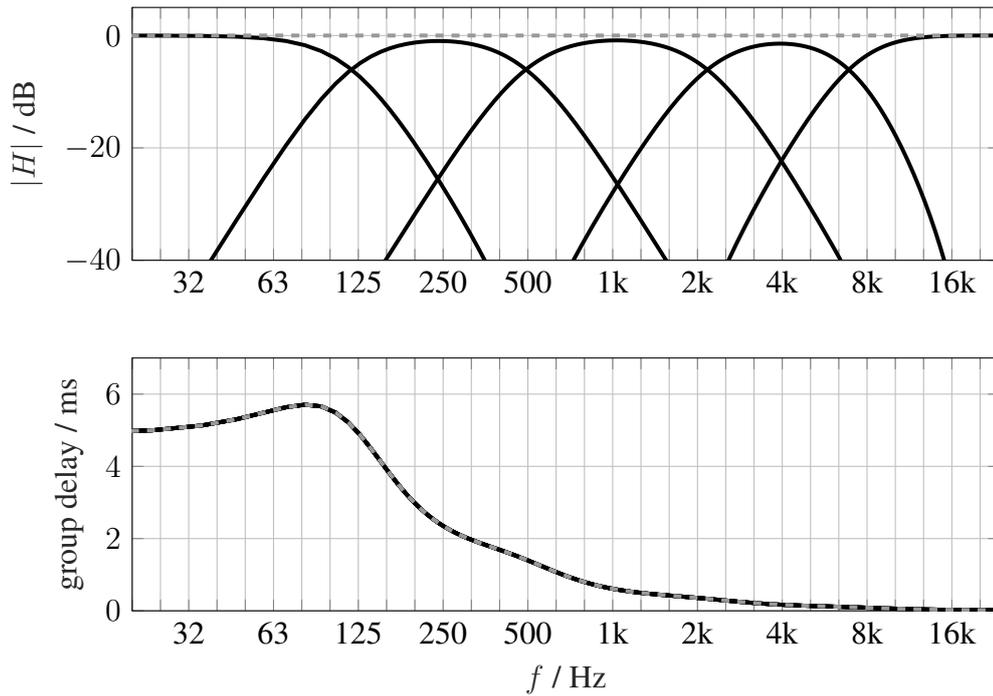


Figure 3 – Amplitude response $|H|$ and group delay of five-way IIR filter bank built of 4th-order Linkwitz-Riley filters. Solid lines are single band responses, dashed lines are summed output responses.

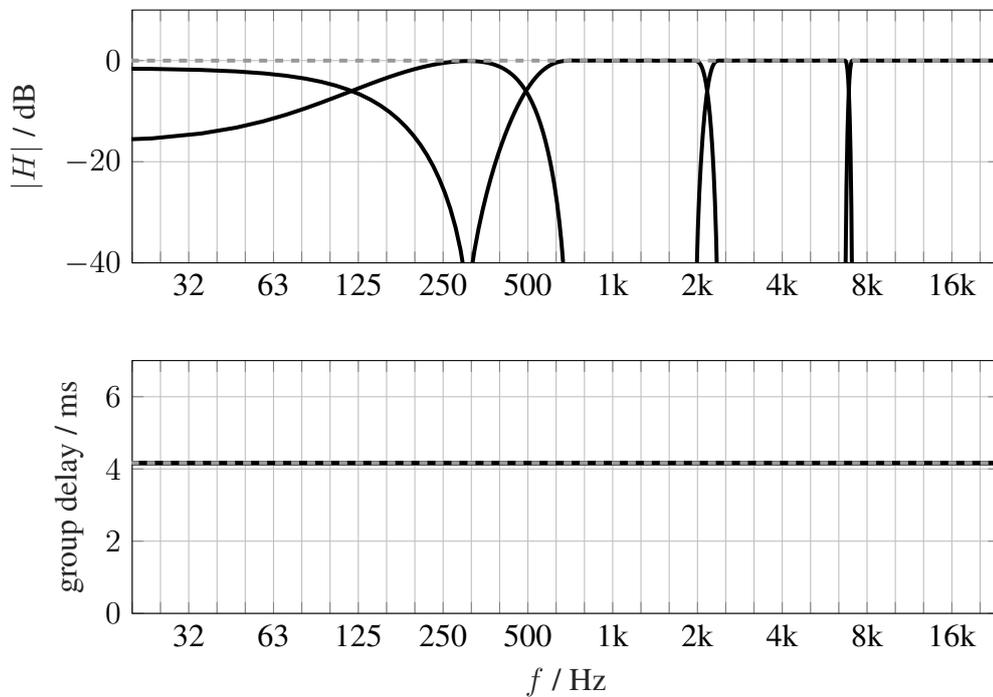


Figure 4 – Amplitude response $|H|$ and group delay of five-way FIR filter bank of order $N = 400$. Solid lines are single band responses, dashed lines are summed output responses.

3 Equalization filters: Measurement-based filter design

3.1 Motivation

Arbitrary first-order polar patterns can be created by a weighted superposition of two different first-order polar patterns covering the whole sphere. In practice, most microphones providing an adjustable polar pattern contain two diaphragms with either one omnidirectional and one figure-of-eight-shaped polar pattern or two opposing cardioid-shaped polar patterns. As it is not possible to construct two perfectly coincident diaphragms, sound sources which are non-equidistant from the two diaphragms will be gathered with a phase shift. After superposition this phase shift leads to constructive and destructive interferences in the resulting frequency response. Further, in some cases one of the two diaphragms gathers a signal with a 180° phase shift with respect to the other diaphragm's signal (e.g. figure-of-eight pattern and source directions around 180°). For low frequencies where the distance of the two diaphragms is small compared to the wavelength, this results in a subtraction of the signals and a decrease in magnitude, whereas at higher frequencies the signals are alternating between being in-phase and out-of-phase, resulting in comb-filtering (cf. figure 5). Additionally, real microphones show direction-dependent and exemplar deviations in the frequency response which generate further deviations in the superimposed frequency response.

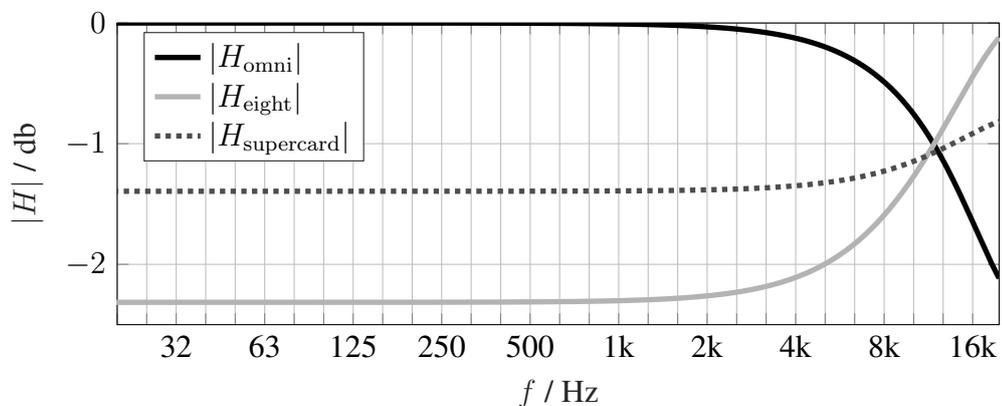


Figure 5 – Simulated magnitude responses of three polar patterns for a source signal from 40° azimuth, created by superposition of two opposing ideal-cardioid polar pattern signals with a diaphragm distance of 1 cm. H_{omni} , H_{eight} and $H_{\text{supercard}}$ correspond to omnidirectional, figure-of-eight and super-cardioid polar patterns, respectively.

If an omnidirectional and a figure-of-eight polar pattern signal are the basis for the arbitrary superimposed first-order polar pattern, the frequency responses of these two signals can be equalized to achieve a polar-pattern-independent constant frequency response. In this work it was decided to maintain the on-axis microphone characteristics, so the resulting polar pattern's frequency response is supposed to resemble the on-axis frequency

response of the microphone. There are two ways to maintain the on-axis microphone characteristics. In the first case the on-axis frequency responses of the omnidirectional and figure-of-eight signal will be equalized to the on-axis response of the microphone. This will be referred to as free-field equalization and will only equalize deviations in the frequency response emerging from the phase shift between signals from non-coincident diaphragms, and due to the non-ideal backwards attenuation if two cardioid diaphragms are used. In the second case the mean frequency response of all directions on a sphere around the microphone, referred to as diffuse-field response (cf. section 3.3), will be equalized to resemble the on-axis frequency response of the microphone. Hence, the diffuse-field equalization not only takes into account the direction-dependent phase shift between the two diaphragm signals, but also the directional variation of the frequency responses of the two microphone diaphragms.

3.2 Free-field equalization

After free-field equalization, the on-axis frequency response is supposed to constantly equal the on-axis response of one individual diaphragm, independent of the synthesized polar pattern. To achieve this goal, the omnidirectional and figure-of-eight on-axis frequency responses are equalized to this on-axis response before being mixed to create arbitrary polar patterns. Figure 6 shows the diaphragm's on-axis magnitude response $|H_{aa,onaxis}|$ and the calculated omnidirectional and figure-of-eight on-axis responses $|H_{aa,omni}|$ and $|H_{aa,eight}|$.

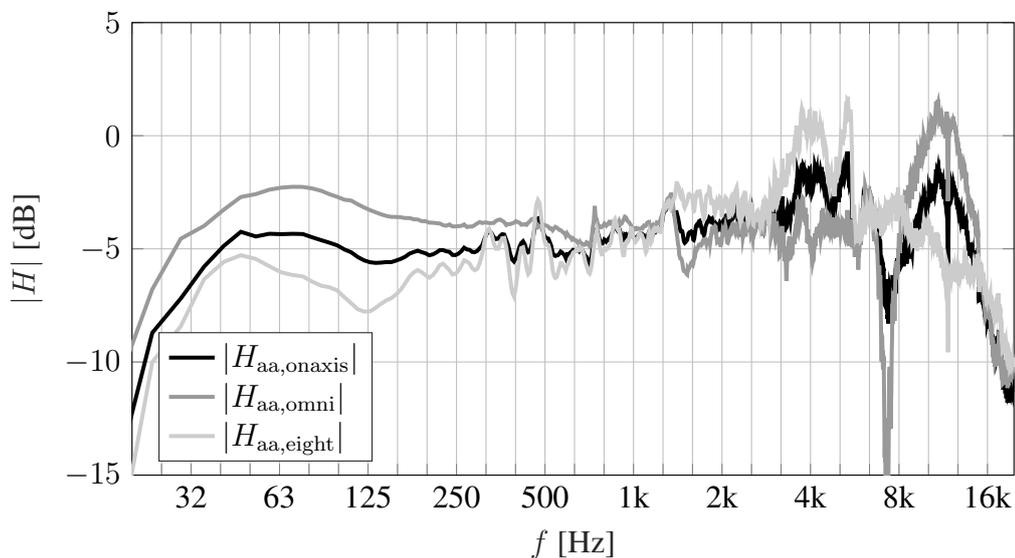


Figure 6 – On-axis magnitude responses of one individual diaphragm $H_{aa,onaxis}$, the synthesized omnidirectional polar pattern $H_{aa,omni}$ and figure-of-eight polar pattern $H_{aa,eight}$.

Now the free-field equalization filters for the omnidirectional and the figure-of-eight polar

pattern $H_{\text{ff,omni}}$ and $H_{\text{ff,eight}}$ can be calculated by division in the frequency domain

$$H_{\text{ff,omni}} = \frac{H_{\text{aa,onaxis}}}{H_{\text{aa,omni}}}, \quad (8)$$

$$H_{\text{ff,eight}} = \frac{H_{\text{aa,onaxis}}}{H_{\text{aa,eight}}}. \quad (9)$$

3.3 Diffuse-field equalization

To calculate the microphone's diffuse-field response, generally, a limited number of measurements is used to approximate the mean frequency response of all directions on a sphere around the microphone. As first-order polar patterns are rotationally symmetric, the frequency response measured for one direction equals the frequency responses for all measurement directions when rotating around the symmetry axis. Further, the mean response for all possible measurement directions on the unit sphere around the measured microphone can be approximated by measurements on a horizontal orbit by weighting the measurements with weights w_i corresponding to the amount of the surface area the spherical segment A_i containing the measurement position represents. With the unit sphere S , the azimuth angle $\varphi = [0, \pi]$ and the polar angle $\vartheta = [0, 2\pi]$, the weight w_i corresponding to the spherical segment A_i bounded by angles φ_i and φ_{i+1} is calculated as

$$w_i = \frac{A_i}{\oint_S dS} = \frac{\int_{\varphi=\varphi_i}^{\varphi_{i+1}} \int_{\vartheta=0}^{2\pi} \sin \varphi \, d\vartheta \, d\varphi}{4\pi} = \frac{-\cos \varphi_{i+1} + \cos \varphi_i}{2}. \quad (10)$$

Note that the angles φ_i and φ_{i+1} do not necessarily describe the measurement positions, but the bounds of the spherical segment the current measurement describes. E.g. for the first measurement of measurement positions in a semicircle, $\varphi = [0, \pi]$, the corresponding segment is limited by 0° and half the angle to the next measurement point. If the horizontal-plane measurements are done in equidistant steps of 10° beginning at $\varphi = 0^\circ$, the first two weights are calculated as

$$w_0 = \frac{-\cos(5^\circ) + \cos(0^\circ)}{2} \approx 0.0019, \quad (11)$$

$$w_1 = \frac{-\cos(15^\circ) + \cos(5^\circ)}{2} \approx 0.0151. \quad (12)$$

By using the weights w_i and 19 impulse response measurements H_i , taken in 10° steps from 0° to 180° , the magnitude diffuse-field response of omnidirectional and figure-of-eight polar patterns are calculated as

$$|H_{\text{aa,df,omni}}| = \sum_{i=0}^{18} w_i |H_{\text{omni},i}|, \quad (13)$$

$$|H_{\text{aa,df,eight}}| = \sum_{i=0}^{18} w_i |H_{\text{eight},i}|. \quad (14)$$

If an equalization filter is calculated from this diffuse-field response, it should be weighted according to the energy the pattern picks up, otherwise the resulting magnitude response will have a magnitude offset with respect to the unequalized polar pattern. In case of a figure-of-eight polar pattern, for example, the diffuse-field equalization filter needs to be weighted by $1/\sqrt{3}$,

$$|H_{df,eight}| = 1/ \left(\sqrt{3} \sum_{i=0}^{18} w_i |H_{eight,i}| \right), \quad (15)$$

as the figure-of-eight polar pattern only picks up $1/3$ of the energy the omnidirectional pattern picks up,

$$\frac{\int_0^{2\pi} \int_0^\pi (\cos \vartheta)^2 \sin \vartheta \, d\vartheta \, d\varphi}{\oint_{\mathbb{S}^2} dS} = \frac{1}{3}. \quad (16)$$

Obviously, in case of the omnidirectional polar pattern, no additional weighting is needed,

$$|H_{df,omni}| = 1/ \left(\sum_{i=0}^{18} w_i |H_{omni,i}| \right). \quad (17)$$

Figure 7 shows the diaphragm's on-axis magnitude response $|H_{aa,onaxis}|$ and the calculated diffuse-field magnitude responses of omnidirectional and figure-of-eight pattern $|H_{aa,df,omni}|$ and $|H_{aa,df,eight}|$.

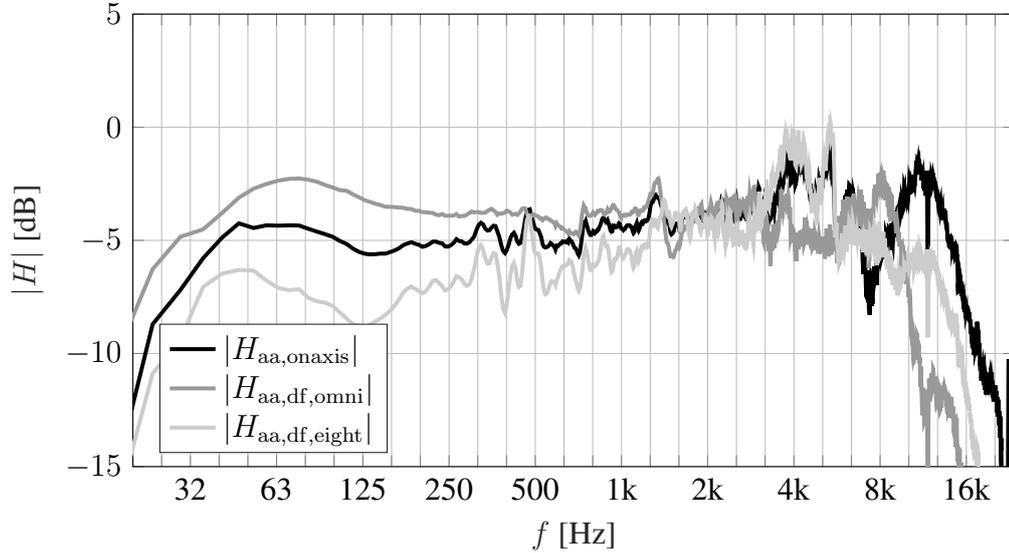


Figure 7 – On-axis magnitude response of one individual diaphragm $H_{aa,onaxis}$ and diffuse-field magnitude responses of the synthesized omnidirectional polar pattern $H_{aa,df,omni}$ and of figure-of-eight polar pattern $H_{aa,df,eight}$.

3.4 1/n-octave band smoothing

These raw equalization filters are then smoothed in 1/3-octave bands, so steep narrow resonances which emerge from the directionally-variant interferences between the two diaphragm's signals and from reflections at the measurement equipment are not accounted for so strongly. An 1/n-octave band smoothing can be accomplished by moving average filtering the magnitude response with a window of length 1/n-octave. As this implies a variable filter length depending on the frequency bin k , the filtering process is defined for each bin. To calculate the smoothed magnitude response $|H_{\text{sm}}[k]|$, the bins of the original magnitude response $|H[k]|$ in the range of an 1/n-octave band are put into the vector \mathbf{h}_k . The lower and upper range limits are defined by $k_{\text{lower}} = k \cdot 2^{-1/(2n)}$ and $k_{\text{upper}} = k \cdot 2^{1/(2n)}$ for each frequency bin k . Still, k_{lower} and k_{upper} have to be rounded to the next integer value and clipped to the range $[1, N/2 + 1]$ with N being the number of discrete Fourier transform coefficients and $N/2 + 1$ resembling the number of positive-frequency bins. To emphasize the current bin k the vector \mathbf{h}_k is weighted by the window function \mathbf{w}_k and the result is divided by the sum of the window function's values to obtain the weighted average

$$|H_{\text{sm}}[k]| = \frac{\mathbf{h}_k^T \mathbf{w}_k}{\sum \mathbf{w}_k}. \quad (18)$$

As window function a Hann window of the same length as the vector \mathbf{h}_k is chosen.

3.5 Minimum-phase filters via cepstral domain

The cepstrum was introduced by Bogert et al. [BHT63] following the idea that an echo of a signal resembles an additive component in the logarithm of the magnitude spectrum. In the discrete-time domain the complex cepstrum $c_x[n]$ of a signal $x[n]$ is defined using the discrete Fourier transform (DFT)

$$c_x[n] = \mathcal{IDFT} \{ \ln(\mathcal{DFT}\{x[n]\}) \} = \mathcal{IDFT} \{ \ln(|X[k]| + j\angle X[k]) \}, \quad (19)$$

with $\ln(\cdot)$ being the complex logarithm. For reconstruction of the signal $x[n]$ the inverse operation is given by

$$x[n] \approx \mathcal{IDFT} \{ e^{\mathcal{DFT}\{c_x[n]\}} \}. \quad (20)$$

As the cepstrum generally is an infinitely long sequence and the inverse discrete Fourier transform generates periodic sequences of length N , the inverse operation does not reconstruct the signal $x[n]$ perfectly due to time aliasing in $c_x[n]$. Still, as the cepstrum decays at least with $1/|n|$, eq. 20 yields a good approximation for long signals.

Any real, causal signal can be expressed as the sum of an even and an odd-symmetric part which after Fourier transform correspond to the real and the imaginary part of the spectrum. Hence, the real part of the Fourier transform of the cepstrum

$$\Re \{ \mathcal{DFT} \{ c_x[n] \} \} = \Re \{ \ln(|X[k]|) + j\angle X[k] \} = \ln(|X[k]|) \quad (21)$$

transforms to the even-symmetric part of the cepstrum $c_x[n]$. This leads to the definition of the real cepstrum $\hat{c}_x[n]$, only using the magnitude of the spectrum $X[k]$ and the real-valued logarithm,

$$\hat{c}_x[n] = \mathcal{IDFT} \{ \ln(|X[k]|) \} . \quad (22)$$

If the input signal $x[n]$ is real and minimum-phase, then the complex cepstrum $c_x[n]$ is causal and hence can be reconstructed from the even-symmetric part which corresponds to the real cepstrum $\hat{c}_x[n]$,

$$c_x[n] = \hat{c}_x[n]l[n], \quad (23)$$

$$l[n] = \begin{cases} 1, & n = 0, N/2 \\ 2, & 1 \leq n < N/2 \\ 0, & N/2 < n \leq N - 1 \end{cases} . \quad (24)$$

Similarly, the minimum-phase signal $x_{\min}[n]$ with the same magnitude spectrum as the non-minimum-phase signal $x[n]$ can be created by taking the real cepstrum $\hat{c}_x[n]$ (eq. 22), reconstructing the complex cepstrum $c_x[n]$ (eq. 23) and then using the inverse cepstral operation (eq. 20),

$$x_{\min}[n] \approx \mathcal{IDFT} \{ e^{\mathcal{DFT}\{\hat{c}_x[n]l[n]\}} \} . \quad (25)$$

3.6 Magnitude responses of the final filters

Figures 8 and 10 show magnitude responses of the final 1/3-octave-smoothed, minimum-phase free-field and diffuse-field equalization filters, respectively. The filters are regularized for very low and very high frequencies to prevent excessive boosts. Figure 9 and 11 show magnitude responses of omnidirectional and figure-of-eight polar pattern after free-field and diffuse-field equalization compared to the target on-axis magnitude response. The equalized magnitude responses follow the target magnitude response very well (except for very low and very high frequencies in the diffuse-field case, where the regularization takes effect) and hence prove the effectiveness of the equalization.

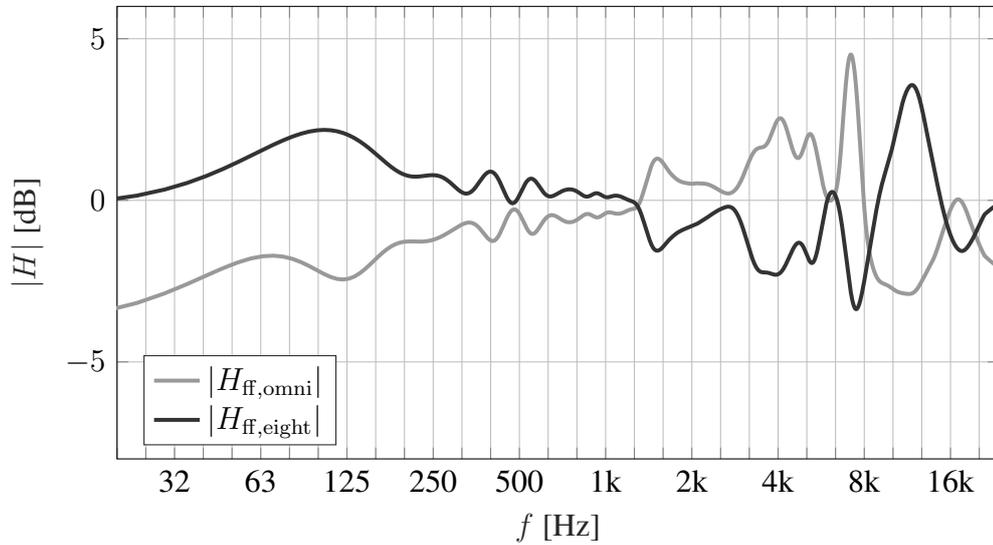


Figure 8 – Free-field equalization filter magnitude response for omnidirectional polar pattern $|H_{aa,ff,omni}|$ and for figure-of-eight pattern $|H_{aa,ff,eight}|$.

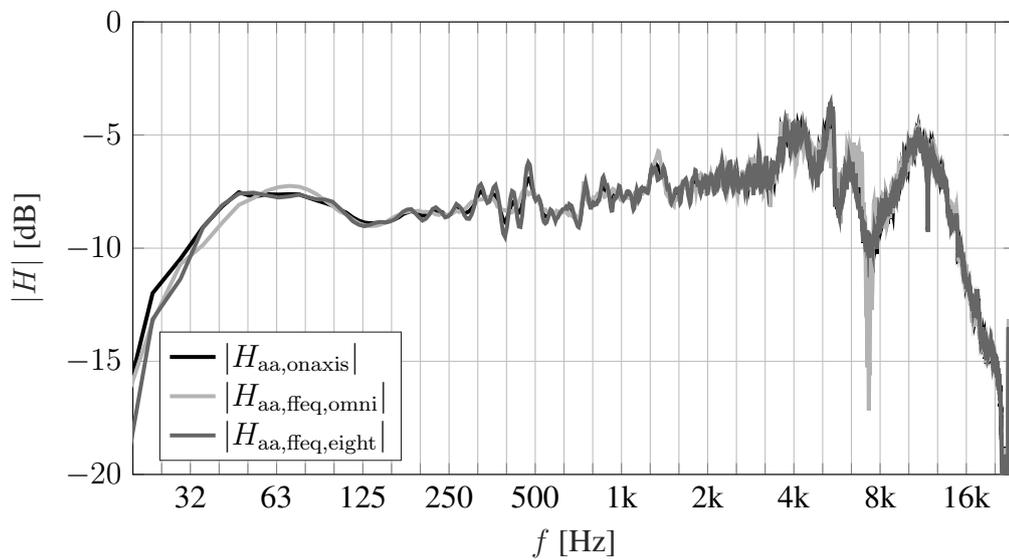


Figure 9 – On-axis magnitude response of one individual diaphragm $|H_{aa,onaxis}|$ and free-field equalized free-field magnitude responses of the omnidirectional polar pattern $|H_{aa,ffeq,omni}|$ and of figure-of-eight polar pattern $|H_{aa,ffeq,eight}|$.

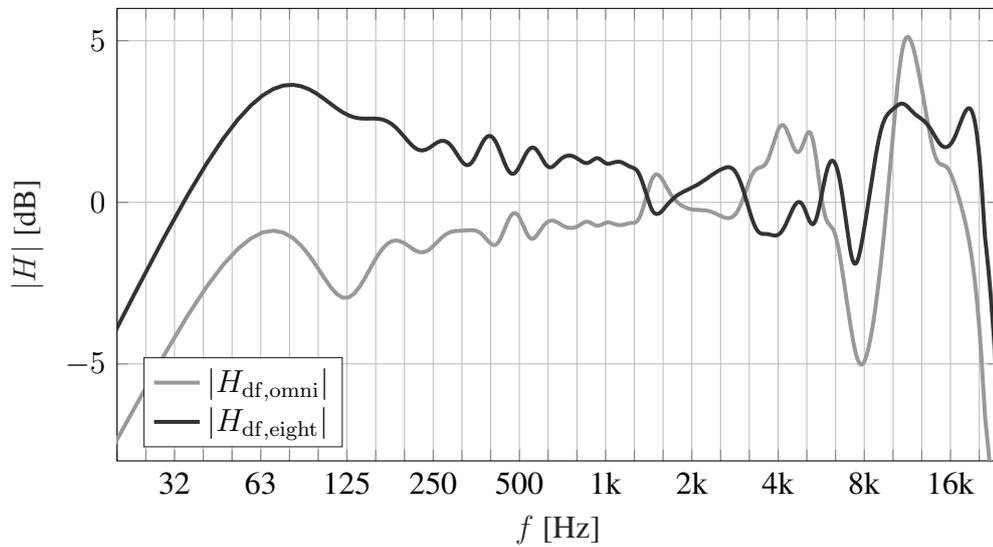


Figure 10 – Diffuse-field equalization filter magnitude response for omnidirectional polar pattern $|H_{aa,df,omni}|$ and for figure-of-eight pattern $|H_{aa,df,eight}|$.

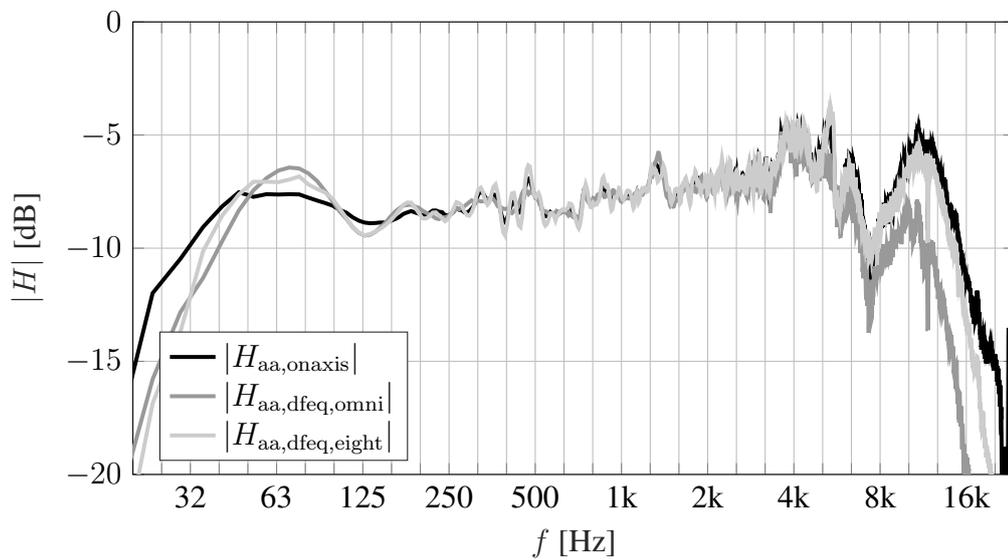


Figure 11 – On-axis magnitude response of one individual diaphragm $|H_{aa,onaxis}|$ and diffuse-field equalized diffuse-field magnitude responses of the omnidirectional polar pattern $|H_{aa,dfeq,omni}|$ and of figure-of-eight polar pattern $|H_{aa,dfeq,eight}|$.

4 Proximity compensation: Model-based filter design

4.1 The proximity effect

In contrast to pressure microphones, pressure-gradient microphones exhibit an increasing low-frequency boost with decreasing source distance. This effect is called the proximity effect and can be undesirable but is also used on purpose to emphasize low frequencies in voice recordings. A compensation filter gives the user the freedom to use any polar pattern and compensate for the proximity effect. Before deriving a compensation filter, the proximity effect is explained mathematically by investigating the pressure gradient. The pressure at distance r created by a point source at the origin, oscillating with frequency ω and amplitude A is described by

$$p = \frac{A}{r} e^{j(\omega t - kr)}, \quad (26)$$

where $k = \omega/c$ is the wave number. For a point source, the radial component is the only non-zero component of the pressure gradient

$$\nabla p = -A \left(\frac{1 + jkr}{r^2} \right) e^{j(\omega t - kr)} \mathbf{u}_r = - \left(\frac{1 + jkr}{r} \right) p \mathbf{u}_r, \quad (27)$$

with \mathbf{u}_r being a unit vector pointing in radial direction. For high frequencies ω or large distances r the term $jk r$ in the numerator becomes large, so the pressure gradient is proportional to $(jk)/r$. For low frequencies ω and small distances r , the term $jk r$ becomes small and can be neglected, so the pressure gradient is proportional to $1/r^2$. Hence, this relation yields an increase of the pressure gradient for low frequencies and close distances and explains the proximity effect. Still, the pressure gradient shows a dependency on frequency ω in far field which is not observed in the output of pressure gradient microphones. Thus, Cotterell [Cot02, p. 58] introduces the equalized pressure gradient, which describes the output of a pressure gradient microphone and loses the frequency dependency and the 90° phase shift. For the output of a mixed pressure, pressure-gradient microphone follows

$$H_p = \left((1 - \alpha) + \alpha \cos(\varphi) \cos(\vartheta) \frac{1 + jkr}{jkr} \right). \quad (28)$$

Here, the factor α defines the ratio between the pressure component and the pressure-gradient component of the microphone as described in section 1.1. The azimuth angle φ and elevation angle ϑ define the direction of the incoming sound wave.

4.2 Continuous-time compensation filter

Figure 12 shows the boost of low frequencies due to the proximity effect described by the transfer function H_p for four distances. To equalize the enhancement of low frequencies

due to the proximity effect, the radially-dependent term in eq. 28 needs to be inverted. In the Laplace domain with complex frequency s , this yields

$$H_{\text{pc}} = \frac{s}{s + c/r}. \quad (29)$$

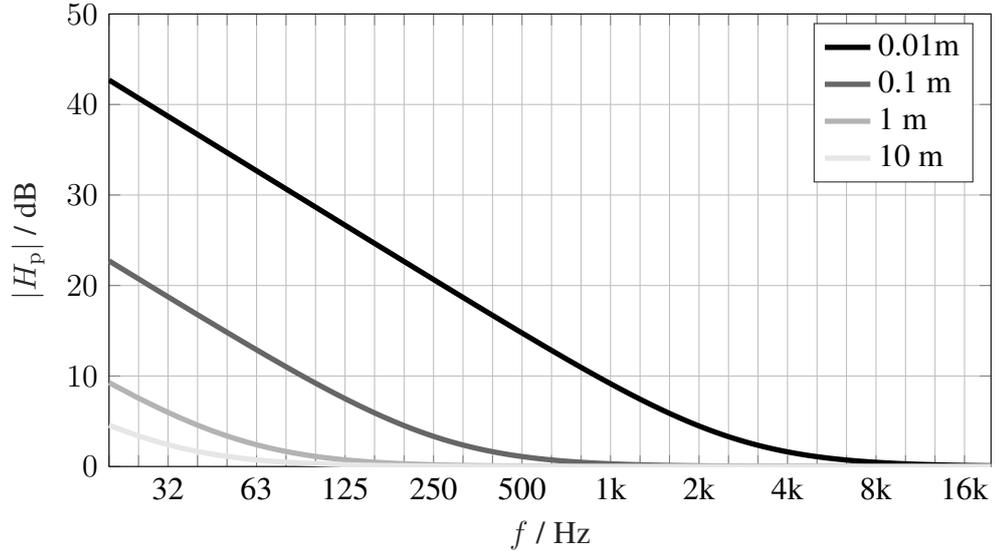


Figure 12 – Magnitude of the transfer function of the bass boost factor H_p due to the proximity effect for a cardioid-directivity microphone and different source distances r .

4.3 Discretization: Comparison of impulse invariance and bi-linear transform

Corrected impulse invariance. The corrected impulse invariance [Mec00][Jac00] method can be used to construct discrete-time filters via sampling of the continuous-time impulse response. Thereby, the part $-\pi.. \pi$ of the imaginary $j\Omega$ -axis in the Laplace s -domain is mapped onto the z -domain unit-circle. Spectral components outside this frequency range, map back onto the unit-circle, resulting in aliasing. Hence, the impulse invariance leads to aliasing if the continuous-time transfer function is not strictly band limited. As the transfer function in eq. 29 shows highpass characteristic, it is not strictly band limited and will lead to aliasing. Still, depending on sampling interval T and on the pole location, the effects of aliasing might be negligibly small.

To obtain a discrete-time transfer function, the transfer function H_{pc} is split into a strictly proper part H_{lp} and a direct throughput $K_{\text{hf}} = \lim_{s \rightarrow \infty} H_{\text{pc}} = 1$,

$$H_{\text{pc}} = H_{\text{lp}} + K_{\text{hf}} = -\frac{1}{sr/c + 1} + 1, \quad (30)$$

which in time-domain corresponds to the impulse response

$$h_{\text{pc}} = h_{\text{lp}} + h_{\text{hf}} = -\frac{c}{r}e^{-c/(r)t} + \delta(t), \quad (31)$$

where $\delta(t)$ represents the Dirac delta function. The discrete-time impulse response is obtained via sampling at discrete time instants $t = nT$,

$$h_{\text{d,pc}} = h_{\text{d,lp}}(nT) + h_{\text{d,hf}}(nT) = -\frac{c}{r}e^{-c/rnT} + \delta(nT). \quad (32)$$

Now, the transfer function in the discrete-time z -domain is calculated as

$$\begin{aligned} \tilde{H}_{\text{d,pc}} &= H_{\text{d,pc}} - \frac{T}{2}h_{\text{d,lp}}(0) = TH_{\text{d,lp,pc}} + K_{\text{hf}} - \frac{T}{2}h_{\text{d,lp}}(0) \\ &= -T\frac{c}{r}\frac{z}{z - e^{-c/rT}} + 1 + \frac{T}{2}\frac{c}{r} \\ &= \frac{(-Tc/(2r) + 1) - e^{-c/rT}(1 + Tc/(2r))z^{-1}}{1 - e^{-c/rT}z^{-1}}, \end{aligned} \quad (33)$$

where $H_{\text{d,lp}} = \mathcal{Z}\{h_{\text{d,lp}}\}$ denotes the z -transformed, discretized impulse response and together with the constant throughput term K_{hf} forms the result $H_{\text{d,pc}}$, which would have been obtained by the (non-corrected) impulse invariance. The subtrahend $\frac{T}{2}h_{\text{d,lp}}(0)$ vanishes for small sampling intervalls T and can be derived by demanding a correct utilization of the Laplace transform at discontinuities [Mec00][Jac00] or convolution invariance [Eit06] for the filter.

Bilinear transform. The bilinear transform [OS10, chapter 7.2.2] avoids aliasing for non-band-limited transfer functions by mapping the infinitely-long imaginary $j\Omega$ -axis in the s -domain onto the unit-circle using a nonlinear transformation

$$s = \frac{2}{T}\frac{1 - z^{-1}}{1 + z^{-1}}. \quad (34)$$

This nonlinear mapping compresses the continuous-time frequency Ω -axis and distorts the resulting discrete-time ω -frequency response including magnitude and phase

$$\Omega = \frac{2}{T}\tan(\omega/2). \quad (35)$$

From this equation, we can see that especially high frequencies are largely distorted by the nonlinear mapping as the tangent is approximately linear for small angles. This distortion can be compensated for by pre-warping of continuous-time characteristic frequencies Ω_i (e.g. cut-off frequencies) to the specified discrete-time frequencies ω_i using eq. 35, if the continuous-time frequency response is piece-wise constant. By inserting eq. 34 in eq. 29, the bilinear-transform discrete-time transfer function $\hat{H}_{\text{d,pc}}$ is calculated as

$$\hat{H}_{\text{d,pc}} = \frac{2}{T}\frac{1 - z^{-1}}{2/T + c/r + (-2/T + c/r)z^{-1}}. \quad (36)$$

Figure 13 shows a comparison of discrete-time proximity compensation filters for a source distance of $r = 2$ cm, derived with conventional impulse invariance $H_{d,pc}$, corrected impulse invariance $\tilde{H}_{d,pc}$ and bilinear transform $\hat{H}_{d,pc}$. Frequency responses obtained via corrected impulse invariance and bilinear transform match the continuous-time response closely. For small distances $r < 1$ cm, the cut-on frequency of this highpass filter increases and aliasing effects for the impulse invariance transfer function, as well as distortion effects for bilinear transform transfer function start to appear. If such very close distances need to be modeled, corrected impulse invariance should be applied after band limitation of the continuous-time transfer function.

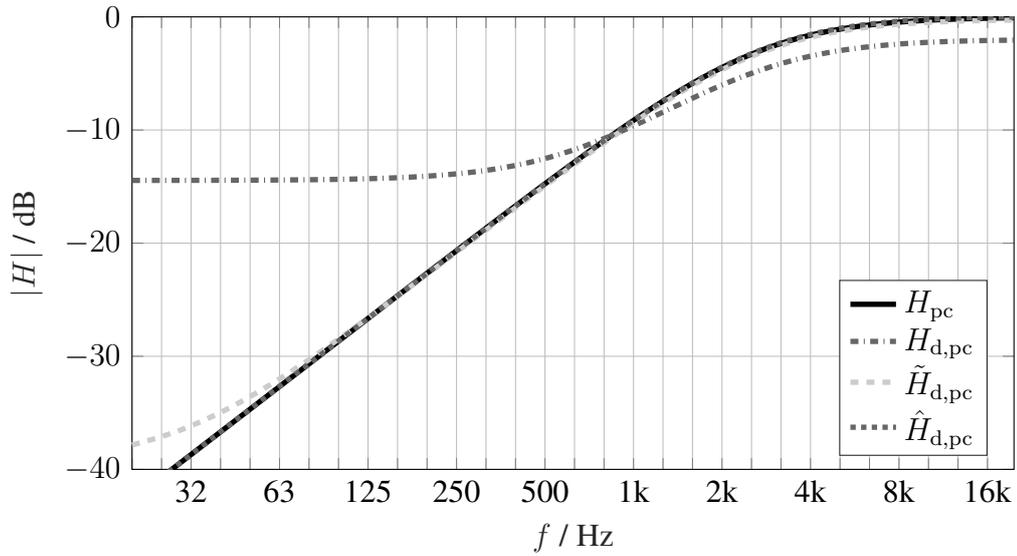


Figure 13 – Proximity effect compensation filters for $r = 2$ cm. Comparison of continuous-time filter H_{pc} , and discrete-time filters $H_{d,pc}$, $\tilde{H}_{d,pc}$ and $\hat{H}_{d,pc}$, obtained via impulse invariance, corrected impulse invariance and bilinear transform, respectively.

4.4 Normalization and comparison to measurements

As the impact of the proximity effect is included in microphone design, i.e. the frequency response of a microphone is designed to be flat for a certain source distance where the proximity effect still has a non-negligible impact, for practical applications it is necessary to normalize the compensation filter with respect to a reference distance r_{ref} ,

$$\tilde{H}_{d,pc,norm} = \frac{\tilde{H}_{d,pc}(r)}{\tilde{H}_{d,pc}(r_{ref})}. \quad (37)$$

Normalization to a distance of $r_{ref} = 1$ m and applying the corrected impulse invariance, yields the discrete-time transfer function of the proximity effect compensation filter

$$\tilde{H}_{d,pc,norm} = \frac{Tc(r-1)/(2r) + 1 - e^{-cT/r}(1 - Tc(r-1)/(2r))z^{-1}}{1 - e^{-cT/r}z^{-1}}. \quad (38)$$

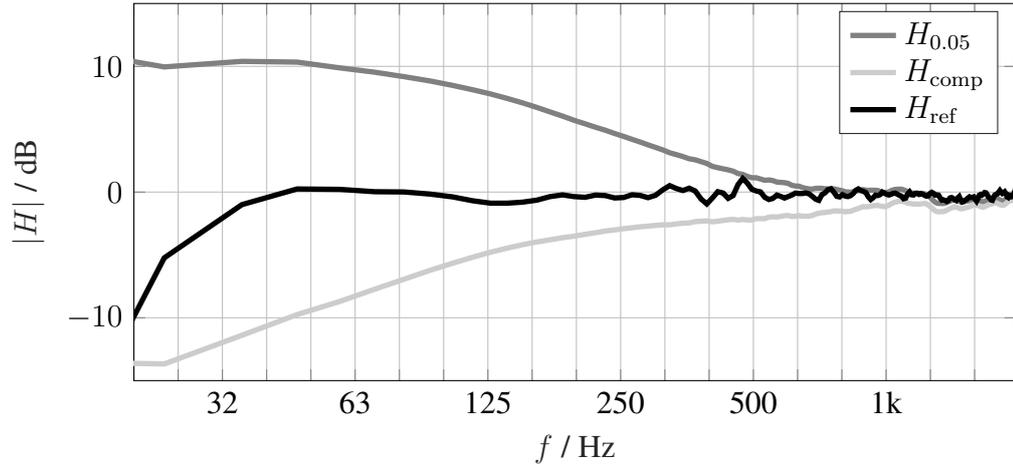


Figure 14 – Measured cardioid-microphone magnitude response $H_{0.05}$ at $r = 5$ cm, H_{ref} at $r_{\text{ref}} = 1$ m and proximity compensated magnitude response H_{comp} .

Figure 14 shows the magnitude responses $H_{0.05}$ of a cardioid microphone measured at a distance of $r = 5$ cm, the amplitude response H_{ref} at reference distance $r_{\text{ref}} = 1$ m and the proximity compensated version H_{comp} using the filter $\tilde{H}_{\text{d,pc, norm}}$. Compared to the measurement, the low-frequency reduction applied by the physically derived filters is too strong. This mismatch is explained by sample variances of the two microphone diaphragms. Although the corresponding frequency responses are calibrated to be very similar, small mismatches in low-frequency magnitude response prevent a boost as strong as in the physical model. In the plug-in implementation (cf. section 6), this problem is solved by letting the user set the distance parameter r by ear, which leads to low-frequency boost for negative r and to low-frequency attenuation for positive r as depicted in figure 15.

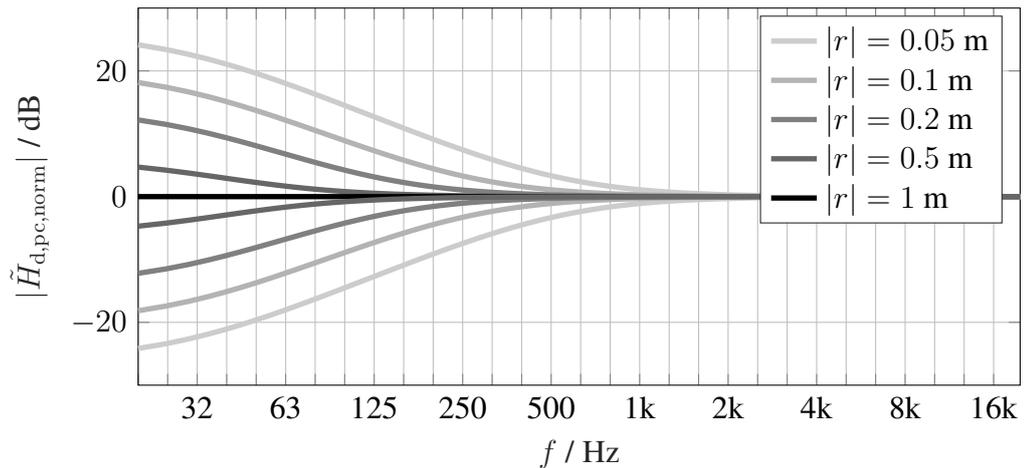


Figure 15 – Proximity compensation filter magnitude responses $|\tilde{H}_{\text{d,pc, norm}}|$ for different distances r . Negative r yield to a low-frequency boost, while positive r yield to low-frequency attenuation.

5 Automatic polar pattern optimization

In many cases the recording engineer adjusts the polar pattern of a microphone with the goal of maximizing a target signal inciding from one direction and simultaneously minimizing unwanted disturber signals inciding from other directions, here called spill. If a recording of target and spill signal is available, as is in post-production or after a short test recording, an algorithm can be used to either maximize the target signal intensity, minimize spill intensity or maximize their ratio. In this work, a grid search algorithm is used to find the intensity-optimal polar pattern. To calculate the intensity I of a first-order polar pattern signal s , composed of omnidirectional signal s_O and figure-of-eight signal s_8 , the individual samples s_i need to be squared and summed. If the calculation is done in blocks, where the block length may vary, it is useful to divide the calculated intensity I_b of block b by the current block length l_b

$$I = \sum_b \frac{I_b}{l_b} = \sum_b \sum_{i=1}^{l_b} \frac{s_{b,i}^2}{l_b} = \sum_b \sum_{i=1}^{l_b} \frac{((1 - \alpha)s_{O,b,i} + \alpha s_{8,b,i})^2}{l_b}. \quad (39)$$

As introduced in section 1.1, α is a weighting factor defining the resulting polar pattern. To avoid conducting the grid search after every block, eq. 39 is expanded as

$$I = (1 - \alpha)^2 \sum_b \sum_{i=1}^{l_b} \frac{s_{O,b,i}^2}{l_b} + \alpha^2 \sum_b \sum_{i=1}^{l_b} \frac{s_{8,b,i}^2}{l_b} + (2\alpha - 2\alpha^2) \sum_b \sum_{i=1}^{l_b} \frac{s_{O,b,i} s_{8,b,i}}{l_b}, \quad (40)$$

allowing to find the weighting factor α for the optimal pattern after signal tracking has ended.

As this algorithm is limited to finding the optimal first-order polar pattern, optimization works better for diverging incidence angles of target and spill signal. This algorithm yields best results if the target signal is in front of the microphone and the spill signal is at least 90° apart. Then it constructs a polar pattern with a zero in direction of the spill, resulting in highest possible spill elimination.

6 Implementation

For creating a plug-in for use in digital audio workstations (DAWs) the C++ framework JUCE¹ is used. JUCE provides wrappers for creating audio plug-ins in different formats (VST, VST3, AU, AUv3, RTAS, AAX) and allows the developer to build these plug-ins from a single codebase for different operating systems. Figure 16 shows the signal flow diagram of the PolarDesigner plug-in. The plug-in expects two inputs from opposing microphone capsules with cardioid-shaped polar patterns. These two signals are added to create an omnidirectional polar pattern in the first channel and are subtracted to create a figure-of-eight polar pattern in the second channel. Then, different types of equalization filters (cf. section 3) can be convolved with the omnidirectional and figure-of-eight signals using uniformly-partitioned convolution in the frequency domain. Due to the use of partitioned convolution no additional delay is introduced at this stage. The next stage consists of the five-channel parallel FIR filter bank described in section 2.3. An FIR filter order of 400 was chosen to enable steep filter slopes even at low frequencies. After an additional gain stage for each of the filter bands, the bands of both patterns are summed to yield the target polar pattern. Finally, the signal is summed over all frequency bands to create a single-channel output signal.

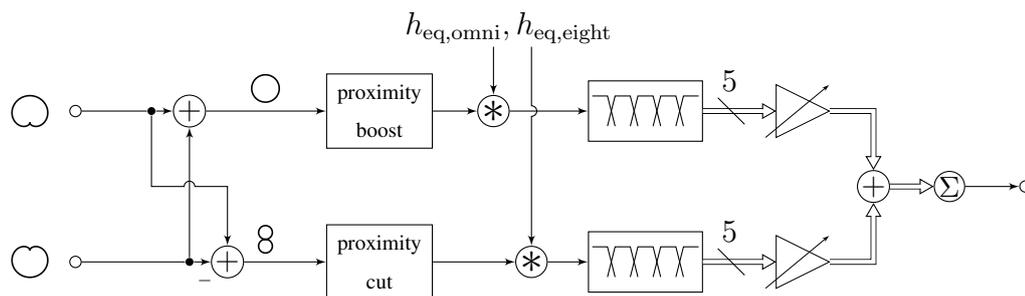


Figure 16 – Signal flow diagram of the plug-in.

Figure 17 shows the user interface of the implemented plug-in. The user has the ability to control the polar pattern in up to five adjustable frequency bands and gets graphical feedback showing the chosen polar pattern. Furthermore, the user can load and save presets, activate free-field or diffuse-field equalization, compensate for the proximity effect, optimize polar patterns based on target and spill signals and synchronize several plug-in instances.

1. juce.com

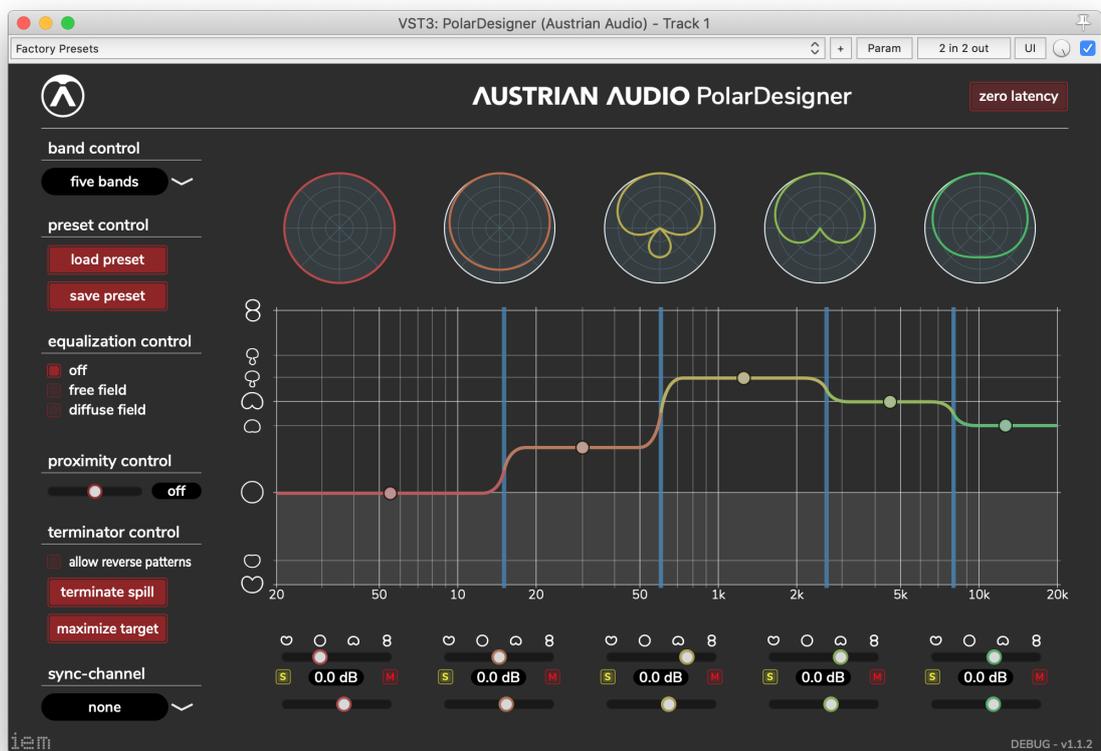


Figure 17 – User interface of the plug-in.

7 Conclusion and Outlook

This work gave an overview of several filter design approaches and their practical application in context of a DAW plug-in for polar pattern control. The plug-in is released open-source under the GPLv3 license and is available on GitHub². Pre-compiled binaries can be downloaded from the Austrian Audio website³.

When using two dual-output dual-diaphragm microphones stacked on top of each other, the polar pattern can be chosen even more freely, allowing an arbitrary rotation in 3D space. This opportunity has been explored in a recently published work [DF20], where a first-order Ambisonics signal is created from a differential array of two dual-diaphragm microphones.

2. <https://github.com/AustrianAudio/PolarDesigner>

3. <https://austrian.audio/>

References

- [BHT63] B. P. Bogert, M. Healy, and J. Tukey, “The quefreny analysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum, and saphe cracking,” in *Proc. Symposium time series analysis*, 1963, pp. 209–243.
- [Bro66] T. Broome, “A frequency transformation for numerical filters,” *Proceedings of the IEEE*, vol. 54, no. 2, pp. 326–327, 1966.
- [Con70] A. G. Constantinides, “Spectral transformations for digital filters,” *Proceedings of the Institution of Electrical Engineers*, vol. 117, no. 8, pp. 1585–1590, 1970.
- [Cot02] P. S. Cotterell, “On the Theory of the Second-Order Soundfield Microphone,” Ph.D. dissertation, University of Reading, 2002.
- [D’A87] J. A. D’Appolito, “Active Realization of Multiway All-Pass Crossover Systems,” *J. Audio Eng. Soc.*, vol. 35, no. 4, pp. 239–245, 1987.
- [DF20] T. Deppisch and C. Frank, “Recording first-order Ambisonics with a differential array of two dual-diaphragm microphones,” *AES 148th Convention*, 2020.
- [Eit06] E. Eitelberg, “Convolution invariance and corrected impulse invariance,” *Signal Processing*, vol. 86, no. 5, pp. 1116–1120, 2006.
- [Jac00] L. B. Jackson, “Correction to impulse invariance,” *IEEE Signal Processing Letters*, vol. 7, no. 10, pp. 273–275, 2000.
- [Lin76] S. H. Linkwitz, “Active Crossover Networks for Noncoincident Drivers,” *AES: Journal of the Audio Engineering Society*, vol. 24, no. 1, 1976.
- [LV05] H. Löllmann and P. Vary, “Efficient non-uniform filter-bank equalizer,” *Proc. of European Signal Processing Conf.(EUSIPCO)*, 2005.
- [LV08] ———, “Low delay filter-banks for speech and audio processing,” *Speech and Audio Processing in Adverse Environments*, pp. 13–61, 2008.
- [Mec00] W. F. Mecklenbräuker, “Remarks on and correction to the impulse invariant method for the design of IIR digital filters,” *Signal Processing*, vol. 80, no. 8, pp. 1687–1690, 2000.
- [OS10] A. V. Oppenheim and R. W. Schaffer, *Discrete Time Signal Processing*, 3rd ed. Pearson, 2010.