Improving Spatial Reproduction by Source Separation

Master Thesis

Nils Meyer-Kahlen

Supervisor: Univ.Prof. Dipl.-Ing. Dr.techn. Alois Sontacchi

Graz, 2019

institut für elektronische musik und akustik

Declaration

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

Nils Meyer-Kahlen
August 2019

Abstract

Technologies for spatial soundfield-capturing and -reproduction have been steadily improved during the last couple of years. The rise of the virtual reality has accelerated the development and has helped Ambisonics to become the dominant format choice. A Higher Order Ambisonics (HOA) production contains detailed spatial information about the sound-sources. Should only a First Order Ambisonics (FOA) or even stereo recording be available, much more rudimentary spatial features are included. This work deals with approaches of extracting individual sound sources in order to create HOA upmixes. To achieve this, a practical ambience extraction approach based on the coherence function is described, which can be used to separate the direct from the diffuse signal part. Furthermore, source separation is used to find and extract sound sources. The studied separation approach relies on the non-negative matrix factorisation (NMF) with multi-dimensional input (also referred to as non-negative tensor factorisation, NTF). Aspects of existing NMF and NTF approaches are being discussed, where the focus lies on their statistical interpretation. A possible algorithm based on such a statistical viewpoint is presented and a multidimensional Gibbs sampler is derived and tested in the audio-application. Apart from this, clustering strategies for the resulting components based on cepstral and spatial features are presented.

Kurzfassung

Technologien zur räumlichen Schallfeldaufnahme, und -wiedergabe wurden in den letzten Jahren stetig weiterentwickelt. Speziell das Aufkommen der virtuellen Realität hat diese Entwicklung beschleunigt und dazu beigetragen, dass sich Ambisonics als dominierendes Format durchsetzen konnte. In einer Higher Order Ambisonics (HOA) Aufnahme ist detaillierte räumliche Information über die Schallquellen enthalten. Liegt allerdings nur ein Stereoformat oder eine First Order Ambisonics (FOA) Produktion vor, ist die Richtungsauflösung deutlich begrenzt. Diese Arbeit beschäftigt sich mit Ansätzen um aus derartigen Aufnahmen einzelne Signalkomponenten zu extrahieren, mit dem Ziel räumlich besser aufgelöste Varianten zu erzeugen. Hierzu wird zunächst ein praktikabler Ansatz zur Trennung von direkten und diffusen Schallanteilen basierend auf der Kohärenzfunktion beschrieben. Darüber hinaus wird Quellseparation verwendet, um einzelne Bestandteile des Signals zu ermitteln und zu extrahieren. Der untersuchte Quellseparationsansatz beruht auf der Nicht-negativen Matrix-Faktorisierung (NMF) mit mehrdimensionalem Eingang (auch Nicht-negative Tensor-Faktorisierung, NTF). Unterschiedliche Aspekte bestehender NMF und NTF Ansätze werden diskutiert, wobei vor allem auf die statistische Interpretation eingegangen wird. Ein möglicher Algorithmus auf Basis eines derartigen statistischen Ansatzes wird vorgestellt und ein Gibbs Sampler für das mehrdimensionale Problem hergeleitet und in der Audioanwendung getestet. Zuletzt werden Gruppierungsstrategien für die extrahierten Signalkomponenten auf Basis von cepstralen und räumlichen Eigenschaften vorgestellt.

Contents

Chapter 1

Introduction

In the field of spatial audio, research has advanced quickly during the recent years. Technologies for playing back spatially distributed sound on flexible loudspeaker setups now are available for many practical situations. Also, binaural renderers, which aim at enabling similar experiences over headphones, are steadily being improved, boosted by the development of the virtual reality. Apart from that, compact loudspeaker arrays for sound projection are being studied. With all these spatial audio technologies at hand, there is one important question left: How should one obtain suitable audio material to exploit them?

Usually, such material is explicitly produced for spatial audio. Techniques include microphone recordings of real sound scenes, for example with Ambisonics microphones, or the creation of sound scenes through artificial spatialization of individual sources. And although a lot of new material is being produced, the vast amount of existing mono and especially stereo music recordings released over the last century is left mostly untouched, due to the unavailability of individual instrument tracks. It is possible to create multichannel versions from mono or stereo recordings in a process referred to as upmixing. Many systems have been proposed, often based on ambience extraction and matrixing. In this work, source separation shall be examined, motivated by the stereo to multichannel upmixing problem.

Furthermore, the virtual reality poses a new challenge for spatial audio rendering. Apart from allowing the listener to turn the head and thereby rotate the sound scene, free movement trough the virtual scene should be enabled. One way of creating such 6 degrees of freedom systems involves interpolation between spatially distributed microphone recordings. Another alternative is to encode individual sources in the 3D space. If the scene was recorded with one or several first-order Ambisonics (FOA) microphones for example, the sources need to be extracted. Besides classical beamforming techniques, source-separation that takes both the time-frequency and the spatial domain into consideration could be a promising alternative.

Blind Source Separation. Blind source separation deals with the problem of extracting components from a mixed signal, when no or only little prior knowledge about the sources is available. It is studied in many different disciplines including image processing, medical diagnostics and finance. The objective of this work is to use blind acoustic source separation algorithms for extracting individual instrument tracks of complete stereo or FOA mixes. What should be noted is that the state of the art sound source separation methods yielding the best separation results are almost exclusively informed approaches, which involve extensive training [SLI18]. Nevertheless, blind approaches relying solely on finding latent structure in the data are still a challenging and interesting topic to study. Also, they were among the technologies used for successfully restoring live recordings from the Beatles' american tour in the 1960s [Cla17], which partly inspired this work.

Approaches for blind sound source separation include streaming methods, commonly adopted in computational auditory scene analysis (CASA), independent component analysis (ICA), which aims at finding statistically independent components, and independent subspace analysis (ISA), which is ICA on the magnitude spectrogram of the mixed signal [LL09]. Beginning with [LS99], non-negative matrix factorization (NMF) was popularized and studied by many researchers during the last 20 years, resulting in an ever growing number of publications, in which many approaches are being presented, evaluated and discussed.

The Present Approach. Many researchers have shown one or the other NMF approach or variant to perform better or worse in simple examples. In this thesis, NMF should be understood as one building block of a complete separation system. Advances in the direction of such multi-stage systems have been done mostly by Fitzgerald [Fit11]. The construction of the system used depends strongly on the specific separation task.

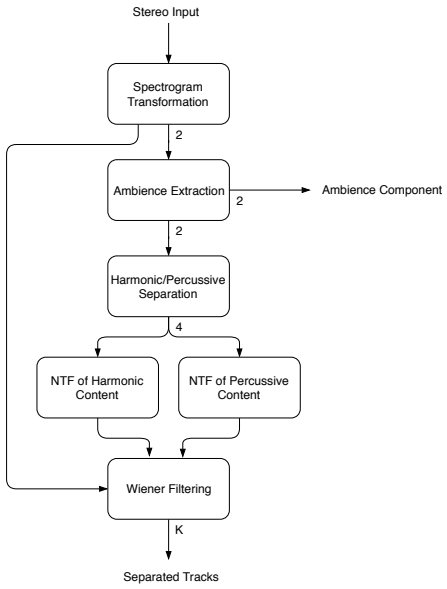


Figure 1.1 – Possible multi-stage separation process.

In such a system, the decomposition task is divided up into several steps. Apart from NMF, two parametric-pre processing steps are described in chapter 3. First of all, ambience extraction, based on [AJ02], could be used to separate stochastic from deterministic signal components. The stochastic components constitute the ambient part of the signal, which can be used in upmixing. The direct signal is then separated into harmonic and non-harmonic components. A simple yet effective algorithm is available [Fit10].

After this, the harmonic and percussive components can be separated using NMF. In some cases, NMF might only be applied to one of the signals, depending on the specific material. In Figure 1.1 the block is called "NTF" instead of "NMF". It stands for "non-negative tensor factorization", which is used for dealing with multichannel input signals. The spatial information can aid the separation progress when taken into account effectively. The main focus of this thesis lies on the statistical formulation of NMF and NTF along with it's solution through full Bayesian inference, which is presented in chapter 4, preceded by a summary of the problem. Whenever NMF or NTF is applied, a number of components, which is typically larger then the number of instruments, is found. Clustering of the components to form instrument signals is mentioned in chapter 6.

With the multichannel algorithms at hand, FOA input data can be processed. Since also the output is intended to be re-panned using Ambisonics, the important fundamentals of this technology are briefly summarised in chapter 5.

All of the described algorithms are based on the spectrogram of the signal, so spectrogram analysis and resynthesis of a time signal are an important part of the system as well. Usually, the short time Fourrier transform (STFT) is used. This work demonstrates that the constant-Q transform can be a very good alternative, especially since perfectly reconstructing inverse transforms are available by now [HDVG13]. The constant-Q transform offers a logarithmic frequency resolution, which is closer to human perception than the linear resolution of the STFT. The underlying concepts of the non-stationary Gabor transform are summarised in chapter 2.

The evaluation of source separation in general and in the context of spatial audio is a challenging field in itself [Roh15]. To exemplary show the effectiveness of the presented algorithms in chapter 7, simple and objective energy-based criteria and considered [VGF06].

Chapter 2

Constant-Q-Transform

For all algorithms presented in this thesis, a time-frequency representation of the input signal is the most important foundation. In particular, the Constant-Q-Transform (CQT) is used, as it can improve many audio specific algorithms due to it's improved time-frequency resolution. When it was first proposed [Bro91], it did not have a functional inverse transformation, making it suitable for analysis, but not for re-synthesis. Later, a more efficient way of processing and a near perfectly reconstructing algorithm was developed in [SK10]. It was shown that this particular implementation can lead to improvements in the separation results of NMF [FJCR11]. Shortly after the publishing of the improved CQT variant and it's implementation, a completly perfectly reconstructing version, based on the mathematical concept of non-stationary Gabor frames was introduced [VHDG11]. The aim of this chapter is to summarise the most important theory and to define the notations used later on in the thesis.

Short Time Fourier Transform. Many signal processing algorithms are based on modifying a signal's spectrogram, which is an important representation, because the frequency content of a signal can be examined over time. This distinguishes it from the classical Discrete Fourier Transform (DFT)

$$\text{DFT}(x[t]) = \underline{x}[f] = \sum_{t=0}^{T-1} x[t] e^{-\frac{i2\pi ft}{T}}, \tag{2.1}$$

where all the information about the temporal structure of the signal $x[t]$ is hidden in the phase of the Fourier coefficients $\underline{x}[f]$. Throughout the thesis, the frequency index shall be called f and the underline will denote complex numbers.

Gabor Transform. The theory behind spectrogram representations goes back to the Gabor Transform [Gab47], which can be defined by

$$\text{STFT}(x[t]) = \underline{x}_{fn} = \sum_{t=0}^{T-1} x[t]w[t - nR]e^{-2\pi i \frac{ft}{T}}, \tag{2.2}$$

$$w[t] = e^{-\alpha^2 t^2}. \tag{2.3}$$

The values $\underline{x}_{fn}$ are arranged in a $(F \times N)$ spectrogram matrix $\underline{\mathbf{X}}$, where N is the number of time instances analysed and F is the number of frequency points. Both are distributed linearly along time and frequency. Every value $\underline{x}_{fn}$ can be thought of as covering a small box in the time-frequency plane.

Gabor in Practise. Inspired by quantum theory, Gabor showed that the area of this time-frequency box is minimal when using a Gaussian function as window. In this case, the Gabor uncertainty turns into an equality.

$$\Delta_f \Delta_t \leq \frac{1}{2} \tag{2.4}$$

For practical application, a window function $w[t]$ which has finite support L is desirable. A DFT will then be calculated with respect to the the non-zero, windowed parts of the signal. In this case, the frequency resolution of each time-slice is limited to $\frac{\nu_s}{L}$, where ν_s is the sampling frequency. The transform with the finite time-support window is

$$\underline{x}_{fn} = \sum_{t=0}^{L-1} x[nR + t]w[t]e^{-2\pi i \frac{ft}{T}}. \tag{2.5}$$

The windowing function can be chosen according to the desired properties of the STFT, since different shapes lead to different leakage of one frequency bin to it's neighbours. A typical choice is the Hann window, which is centred around zero and given by

$$w[t] = \begin{cases} \cos^2(\frac{\pi t}{L}) & |t| < \frac{L}{2} \\ 0 & \text{else.} \end{cases} \tag{2.6}$$

For a practical algorithm, which follows from 2.5, a causal definition is used

$$w[t] = \begin{cases} \frac{1}{2} - \frac{1}{2}\cos(\frac{2\pi t}{L}) & 0 < t < L \\ 0 & \text{else.} \end{cases} \tag{2.7}$$

Inversion. In countless applications of the spectrogram within speech or music processing, modified spectrograms are produced, which need to be inverted in order to obtain a modified time signal. Typically, inversion is done by inverse Fourier transformation and the overlap and add procedure.

If we consider $x_n[t]$ to be the inverse DFT of the n-th time instance, obtained by

$$x_n[t] = \frac{1}{L} \sum_{f=0}^{F-1} \underline{x}_{fn} e^{2\pi i \frac{ft}{T}}, \tag{2.8}$$

reconstruction with the Overlap-and-Add Procedure can be achieved from

$$x[t] = \frac{\sum_{n=0}^{N-1} x_n[t - nR]}{\sum_{n=0}^{N-1} w[t - nR]}. \tag{2.9}$$

This expression is simplified, if the window fulfils the constant overlap and add property

$$\sum_{n=0}^{N-1} w[t - nR] = C. \tag{2.10}$$

Using the Hann-Window the hopsize equal to $R = \frac{L}{2}$, this condition is met with $C = 1$.

Non-Stationary Gabor Transform. In order to compare this transform to the frame-based non-stationary case and in particular the CQT presented in [VHDG11], it can be written down in terms of a collection of NF time-frequency atoms, which are windows, modulated by the complex exponentials. Gabor calls these atoms "logons". Other than in equation 2.5, where the windows are causal and small blocks of the signals are cut out, in this notation, the window is centred around zero and shifted along the time-axis. For the classical STFT denoted above, the corresponding atoms are

$$\underline{\varphi}_{fn}[t] = w[t - nR] e^{-2\pi i \frac{ft}{T}}, \tag{2.11}$$

and the transform can be written as

$$\underline{x}_{fn} = \sum_{t=0}^{T-1} x[t] \underline{\varphi}_{nf}[t]. \tag{2.12}$$

For the stationary Gabor case, these atoms are spaced equidistantly along time at nR and equidistantly along the frequency at $\frac{\nu_s}{L}$. For adaptive spacing in time, one would manipulate the hop-size R. This can for example be used for more accurate resolution of transient sounds [BDJ+11]. For the CQT, adaptivity in frequency is desired. To achieve this, we will first of all, instead of choosing a window of finite support in the time domain, choose a band-limited window in the frequency domain $\underline{\psi}_f$, which has the support L_f. The index f indicates that it will have a different support for each frequency band.

Since the atoms are designed to have finite support in the frequency domain, they have infinite support in the time domain. Consequently, the designed atoms are best applied in the frequency domain, namely to the DFT of the entire signal (with respect to frequency variable f') or on the DFT of larger signal blocks, with their length depending on the atom with the smallest frequency support (termed "sliced CQT", [HDVG13]).

$$\underline{x}_{fn} = \sum_{f'=0}^{F'-1} \underline{x}[f']\underline{\psi}_{fn}[f'] \tag{2.13}$$

For the CQT, the windows will have their center frequencies ν_f logarithmically spaced along the frequency axis, mirrored at the sampling frequency ν_s. The constant B determines how many bins will be placed within each octave

$$\nu_f = \begin{cases} 0 & f = 0 \\ \nu_{min} 2^{\frac{f-1}{B}} & f = 1, ..., F \\ \frac{\nu_s}{2} & f = F+1 \\ \nu_s - \nu_{2F+2-f} & f = F+2, ..., 2F+1. \end{cases} \tag{2.14}$$

The Q-factor is defined as the ratio of center-frequency to bandwidth, or equivalently it's support in the frequency domain

$$Q_f = \frac{\nu_k}{L_f} = const. \tag{2.15}$$

In order to keep the Q-factor constant at all frequencies, the support of the windows should be chosen according to

$$L_f = \begin{cases} 2\nu_{min} & f = 0 \\ \frac{\nu_f}{Q} & f = 1, ..., F \\ \nu_s - 2\nu_F & f = F+1 \\ \frac{\nu_{2F+2-f}}{Q} & f = F+2, ..., 2F+1. \end{cases} \tag{2.16}$$

This placement has some major advantages over the linear scale of the STFT, such as it's closer match to human perception. Particularly, in musical signals, a lot of information is in the lower frequency range, where the fundamental frequencies of many instruments and the voice are located. In [VHDG11], translated and dilated Hann windows (cf. eq.2.6) are used as window functions for $f = 1, ..., F, F+1, ..., 2F+2$. They are given by

$$\psi_f[f'] = w\Big[\frac{\frac{f'\nu_s}{F'} - \nu_f}{\nu_{f+1} - \nu_{f-1}}\Big], \tag{2.17}$$

where f' is the index of the large DFT.

For $f = 0$ and $f = F + 1$, Tukey windows are used in order to cover the frequency range between 0 and ν_{min}, as well as $\nu_{min} 2^{\frac{F-1}{B}}$ and $\frac{\nu_s}{2}$, respectively. As alternative to the constant Q-factor, it can also be desirable to increase the time-resolution at lower frequencies, for example to represent the ERB scale, which becomes linear at low frequencies. In [SK10], this is done by the parameter γ.

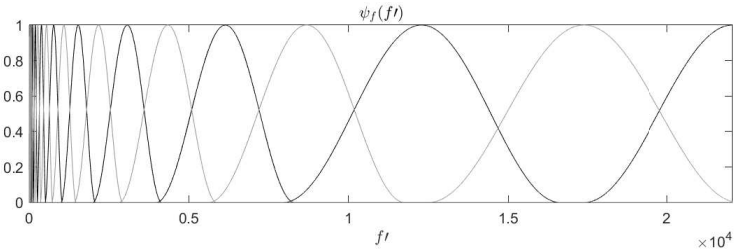


Figure 2.1 – Frequency atoms, which are applied to the DFT of the entire signal, $\nu_{min} = 6, B = 2$.

One window function is available for every frequency band now. If they were applied to the entire signal in the frequency domain, one coefficient for every frequency would be obtained. In the present case of a time-frequency representation, a coefficient for every frequency and every time slice should be obtained, this means that a variant of every frequency window for every time instance has to be created.

The important fact is that to ensure that no information is lost, and ultimately to ensure inverteability, the hop-sizes in the time domain must be adaptive as well. Depending on the support of the frequency window, they should be chosen to fulfil the important support condition

$$R_f \leq \frac{1}{L_f}. \tag{2.18}$$

It can be shown that in this case the set of atoms is a frame [Doe01] and thereby an inverse transformation must exist. This condition needs to be translated to the frequency domain, were the windows are being designed. A time-shift of the window in the time domain is equivalent to a phase-shift in the frequency domain, so the entire set of windows is obtained by

$$\underline{\psi}_{fn}[f'] = e^{-2\pi i \frac{nR_f}{T}} \psi_f[f'] \quad n = 0, ..., N. \tag{2.19}$$

If the support condition is fulfilled, a perfect inverse is found using the atoms $\underline{\gamma}_{fn}$, which form the dual frame. They are given by

$$\underline{\gamma}_{fn}[f'] = \frac{\underline{\psi}_{fn}[f']}{\sum_f \frac{1}{R_f} |\psi_f|^2}. \tag{2.20}$$

The inverse transform is obtained by

$$\underline{x}[f'] = \sum_{f=0}^{2F+1} \sum_{n=0}^{N} \underline{x}_{fn} \underline{\gamma}_{fn}[f'], \tag{2.21}$$

with only an inverse DFT left, to obtaining a signal in the time domain.

Chapter 3

Parametric Pre-Processing

Before engaging in more advanced source separation techniques, ambience separation based on [AJ02] was implemented. Originally, it was intended as a stereo to discrete multichannel upmixing tool. Here it may serve several purposes. Apart from being used as a pre-processing step for later source separation, it is effective on it's own for stereo or FOA to HOA upmixing, where the separated ambience can be spatialized to improve immersion. For this application, decorrelated copies of the ambience component can be created using [CDA18]. A scheme for processing multiple pairs of input channels is presented.

3.1 Signal Model

The ambience extraction method is based on a simple convolutive mixture model. The signal for each of the M channels can be expressed as

$$x_m[t] = \sum_{k=1}^{K} s_k[t] * h_{km}[t] + n_m[t] \tag{3.1}$$

$$= \sum_{k=1}^{K} s_k[t] * d_{km}[t] + \sum_{k=1}^{K} s_k[t] * r_{km}[t] + n_m[t]. \tag{3.2}$$

Imagine a mixture of K sources $s_k[t], k = [1, ..., K]$, where each source has a propagation path to each of a set of M receivers. The receivers could either be thought of as actual microphones in a room or channels of an artificial mix. In the typical application presented here, stereo mixes ($M = 2$) or FOA mixes ($M = 4$) are considered, in which either monophonic sources have been panned, mixed and artificial reverberation has been added, or in which microphones in a room have captured the sound sources. No matter if artificial or real, reverberation should be decorrelated between the channels. The propagation paths can be described by a set of impulse responses $h_{km}[t]$, which consists of a direct and a reverberant part $h_{km}[t] = d_{km}[t] + r_{km}[t]$. An additional portion of background noise $n_m[t]$ is considered, e.g. stemming from an audience in a live recording or the recording equipment itself, as seen in older material.

3.2 Ambience Separation

Coherence Function. The fact that $r_{km}[t]$ is a several hundred milliseconds long impulse response, which is different between the channels, leads to the low correlation of ambient signal components in relation to direct component. The algorithm aims at distinguishing time-frequency regions of high coherence between the channels, which are enhanced to retrieve the direct component, from regions of low coherence, which constitute the ambient part of the signal. The measure is based on the coherence function

$$\underline{c}_{fn}^{(12)} = \frac{\underline{\phi}_{fn}^{(12)}}{\sqrt{\phi_{fn}^{(11)}\phi_{fn}^{(22)}}}, \tag{3.3}$$

where $\phi_{fn}^{(11)}$ and $\phi_{fn}^{(22)}$ are elements of the power spectral densities (PSDs) of $s_1[t]$ and $s_2[t]$ respectively and $\underline{\phi}_{fn}^{(12)}$ are elements of the cross spectral density (CSD). For signals with time-varying statistics such as speech or music, these quantities are approximated by averaging the spectrogram, as done in an IIR fashion in [AJ02]

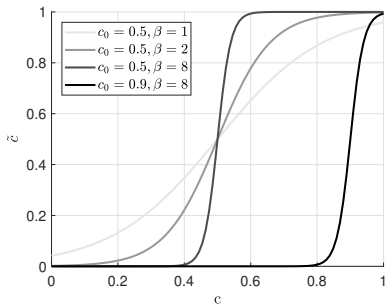
$$\underline{\phi}_{fn}^{(i,j)} = \lambda\underline{\phi}_{f,n-1}^{(i,j)} + (1-\lambda)\underline{x}_{fn}^{(i)}\underline{x}_{fn}^{*(j)}, \tag{3.4}$$

where $\lambda \in [0, ..., 1]$ is the "remembrance" and controls the amount of averaging.

Regularisation. Regularisation is carried out by adding a small constant δ, which is mutltiplied with the mean energy of each frequency band, to the denominator. In this way, small coherences are assigned to low energy bins. If the first signal is omnidirectional as suggested below, it is sufficient to take the mean energy of this signal for regularisation

$$\underline{c}_{fn}^{(12)} = \frac{\underline{\phi}_{fn}^{(12)}}{\delta\sum_n \phi_{nf}^{(11)} + \sqrt{\phi_{fn}^{(11)}\phi_{fn}^{(22)}}}. \tag{3.5}$$

Mapping and masks. To create time frequency masks for the direct and ambient component, a non-linear mapping is applied to the absolute value of the coherence estimation. For this, the hyperbolic tangent is well suited. It has been found that the best results are obtained with a value of c_0 close to 1. This means that only bins with very high coherence are assigned to the direct component. The result of the mapping is used as the mask for the direct component, a mask for the ambience component is created by using $(1 - \tilde{c}_{fn})$. The masks are applied on left and right channel separately, so that the output again is stereo.



$$\tilde{c}_{fn} = \frac{1}{2} - \frac{1}{2}\tanh(\beta\pi(c_0 - |\underline{c}_{fn}^{(12)}|))$$

Figure 3.1 – Mapping function for different values of c_0 and β.

Example. As a testfile, a short sample with a drumset and an organ, convolved with a short room impulse response is used. Coherence estimation with the parameters indicated below yields the following result

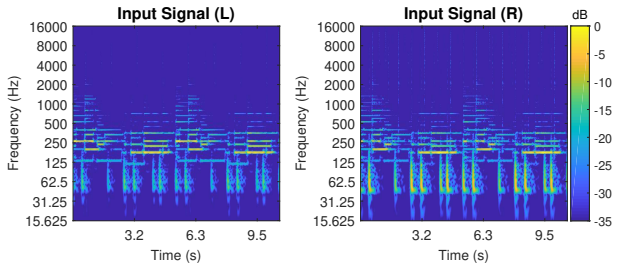
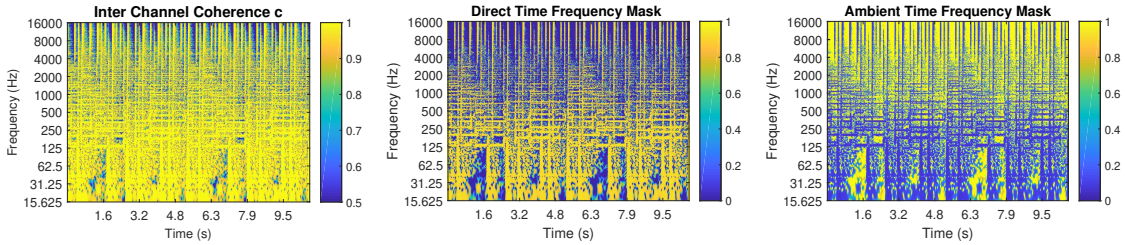


Figure 3.2 – CQT spectrogram of the stereo input signal, consisting of drums and organ convolved with a short room impulse response to add reverberation. The drums are slightly panned to the right, the organ to the left.



(a) (b) (c)

Figure 3.3 – (a) Coherence before mapping, (b) mapping to the direct component, and (c) mapping to the ambient component. Regions with active instruments have a high coherence. Decay processes, such as after the hi-hat beats, exhibit a lower coherence. The parameters were set to $\lambda = 0.8$, $\delta = 10^{-4}$, $c_0 = 0.95$ and $\beta = 8$.

3.2.1 Panning Index

From the same coherence based processing it is possible to create a so called "panogram", based on the panning index [AJ04]. If sources are panned away from the center, but not all the way to one side, it can already be possible to achieve successful source separation, when masks based on the panning regions are used.

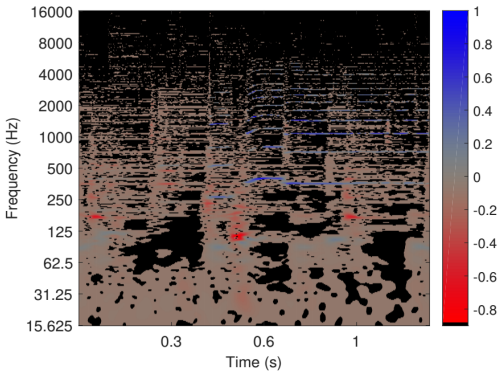


Figure 3.4 – Panning index of the direct signal component. This is a classic jazz recording, where the saxophone is panned to one side. The saxophone with it's rich overtone structure can be seen in the thin blue lines starting with the first sax note at approximately 0.5 seconds. Scaling is done according to the component panned furthest to one side.

3.2.2 Multichannel Scheme

In an environment with more than two channels, this approach can be extended to measure the pair-wise coherence. It has been found, that measuring the pairwise coherence between the omnidirectional signal (i.e. the sum of all signals) and the directional channels is most effective. In Ambisonics, the directional channels are obtained by decoding to a suitable layout, cf. chapter 5. If a source is panned to one of the directions, the directional signal will be strongly correlated with the omni channel and the time-frequency bins will be assigned a high coherence. The T-F bins for which there is no source present in the directional channel will have a lower correlation with the omnidirectional part. This means that for the parts with no source present, mostly ambience signal is obtained, i.e. the reverberation of the other far-panned instruments. Direct and ambient masks are created for each directional channel as described above.

3.3 Harmonic/Percussive Separation

As an additional step in the separation procedure, harmonic/percussive separation is implemented. When comparing different approaches, also based on time-adaptive non-stationary Gabor transform, it was found that a simple heuristics based on median filtering [Fit10] gives very good results.

This approach is based on the non-adaptive STFT, as in this case a constant resolution over time and frequency is important. Two spectrograms are created from every input channel by applying median filters. For the percussively enhanced spectrogram, a median filter of length L_p is along the rows $\mathbf{x}_n^T$ of the magnitude spectrogram and for the harmonically enhanced spectrogram a median filter of length L_h is applied to the columns $\mathbf{x}_f$

$$\mathbf{p}_n = \text{median}(\mathbf{x}_n^T, L_p), \qquad \mathbf{h}_f = \text{median}(\mathbf{x}_f, L_h). \tag{3.6}$$

From these filtered spectrograms, masks are created and applied to the complex input spectrogram in the sense of a Wiener filter

$$\underline{x}_{fn}^{(p)} = \underline{x}_{fn} \frac{p_{fn}}{p_{fn} + h_{fn}}, \qquad \underline{x}_{fn}^{(h)} = \underline{x}_{fn} \frac{h_{fn}}{p_{fn} + h_{fn}}. \tag{3.7}$$

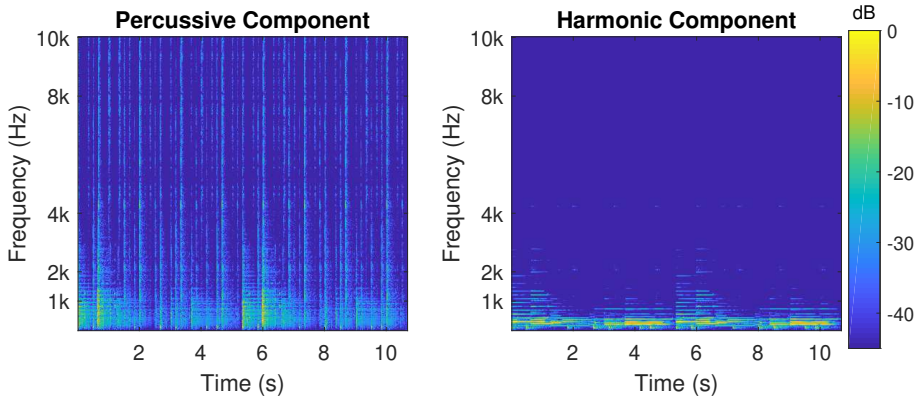


Figure 3.5 – Result of a harmonic/percussive separation.

Chapter 4

Non-Negative Matrix Factorization

4.1 Standard NMF

When NMF was first introduced and popularized it was understood as an approach for low-rank model approximation by constrained optimization [LS99].

A data matrix $\mathbf{X} \in \mathbb{R}_+^{F \times N}$ can be decomposed into a product of matrices in many different ways. In matrix factorization (MF) the aim is to factorize the $\mathbf{X}$ into a product of two matrices $\mathbf{W} \in \mathbb{R}_+^{F \times K}$ and $\mathbf{H} \in \mathbb{R}_+^{K \times N}$. Rank reduction is achieved by setting $K \ll \max(F, N)$.

$$\mathbf{X} \approx \mathbf{WH} = \mathbf{Y} \tag{4.1}$$

In the context of audio source separation, the data matrix $\mathbf{X}$ is either the magnitude spectrogram $|\underline{\mathbf{X}}|$ or the power spectrogram $|\underline{\mathbf{X}}|^2$ of an audio signal. The complex spectrogram $\underline{\mathbf{X}}$ is often times computed by the STFT or, as in this work, by the CQT. The magnitude or power spectrogram only has non-negative entries. In our model, separate uncorrelated sound sources are mixed to form this spectrogram, only allowing for additive combinations. Thus no negative entries of $\mathbf{W}$ and $\mathbf{H}$ are allowed either and the term non-negative matrix factorization (NMF) is established.

NMF has the inherent property of finding K re-occurring components. In musical source separation, these often represent different notes, possibly played by different instruments. The columns of $\mathbf{W}$ correspond to spectral templates for these note events and $\mathbf{H}$ characterizes their activation over time.

When viewing NMF as optimization problem, the factorization is carried out by iteratively minimizing a function $D(\mathbf{X}, \mathbf{WH})$ under the non-negativity constraint. As a cost functions, different measures for the dissimilarity of two matrices are used.

$$\min_{\mathbf{W},\mathbf{H} \geq 0} D(\mathbf{X}, \mathbf{WH}) \tag{4.2}$$

A simple example. As an example, the result of the NMF algorithm for the first two bars of "Day Tripper" by the Beatles is presented. Figure 4.1 shows the input magnitude spectrogram matrix and the two factor matrices $\mathbf{W}$ and $\mathbf{H}$. In the "dictionary" matrix $\mathbf{W}$ the columns are the spectra of the separated components. They exhibit a strong harmonic structure, since the components correspond to single notes played on a guitar.

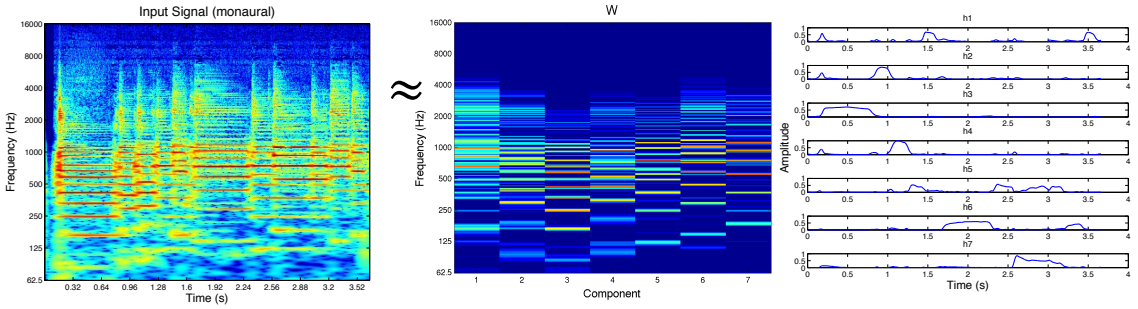


Figure 4.1 – $\mathbf{X} \approx \mathbf{WH}$. The components of the "dictionary" matrix $\mathbf{W}$ clearly have harmonic spectra, as they correspond to single notes played on a guitar.

The rows of the "activation" matrix $\mathbf{H}$, denoted as h1-h7, are shown below, along with the musical notes of the short melody. It is easy to associate the occurrence of some notes with the activation of components, but it is also visible that components are not necessarily activated exclusively. The overtones of one guitar stroke also activate components other then the one of the associated note, cf. h3. The algorithm has no knowledge about the harmonic structure of the content or our desire of being able to match notes to single activated component. It surely is imaginable that improvements will be necessary, especially in more complex acoustic scenarios.

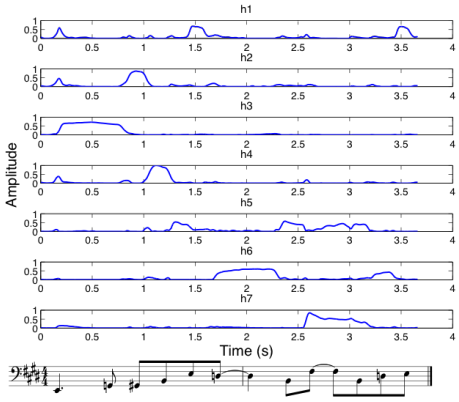


Figure 4.2 – "Activation" matrix together with the score of the excerpt. For some notes it is easy to recognize a matching component, e.g. component h6 and note D2

For this first example, a simple multiplicative update rule minimizing the Euclidean distance was used. The matrices were initialized with uniformly distributed random data. In the following section an overview of cost function, constraints and the statistical viewpoint will be given.

4.1.1 Cost Functions

The cost function $D(\mathbf{X}, \mathbf{WH})$ is obtained by summing up a scalar cost function $d(x, y)$ for every matrix entry over time and frequency

$$D(\mathbf{X}, \mathbf{WH}) = \sum_{f=0}^{F-1} \sum_{n=0}^{N-1} d(x_{fn}, y_{fn}). \tag{4.3}$$

Over several years of NMF research, many cost functions comprising different properties have been proposed. The pioneering paper about cost functions and update rules [LS01] featured two measures: the squared Euclidean distance d_{Euc} and the Kullback-Leibler divergence d_{KL}. Also the Itakura-Saito divergence d_{IS} has been introduced for NMF. In [FBD09] advantageous properties such as it's scale-invariance are pointed out.

$$d_{Euc}(x, y) = \frac{1}{2}(x - y)^2 \tag{4.4}$$

$$d_{KL}(x, y) = x\log\frac{x}{y} - x + y \tag{4.5}$$

$$d_{IS}(x, y) = \frac{x}{y} - \log\frac{x}{y} - 1 \tag{4.6}$$

Later, researchers have focussed on finding general divergences, which encompass the former mentioned, in order to understand their connection and for allowing to adapt derived update rules parametrically. One of the most general descriptions is given by the class of Bregman divergences.

Bregman Divergence. The Bregmann divergence is defined in terms of a strictly convex function $\phi(x)$ that has a continuous derivative

$$d_\phi(x, y) = \phi(x) - \phi(y) - \frac{d}{dy}\phi(y)(x - y). \tag{4.7}$$

For the choice of ϕ shown below, it is simple to show continuity in terms of β and differentiability in terms of x, cf. [HDB11]. When defining ϕ_β in the following way, the class of β-divergence d_β is obtained

$$\phi_\beta(x) = \begin{cases} -\log(x) + x - 1 & \beta = 0 \\ x\log(x) - x + 1 & \beta = 1 \\ \frac{x^\beta}{\beta(\beta-1)} - \frac{x}{\beta-1} + \frac{1}{\beta} & \text{otherwise} \end{cases}, \tag{4.8}$$

$$d_\beta(x, y) = \begin{cases} \frac{x}{y} - \log(\frac{x}{y}) - 1 & \beta = 0 \\ x\log(\frac{x}{y}) + (y - x) & \beta = 1 \\ \frac{x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}}{\beta(\beta-1)} & \text{otherwise} \end{cases}. \tag{4.9}$$

The β-divergence incorporates all the above mentioned measures. When choosing $\beta = 0$ we obtain d_{IS} and for $\beta = 1$ we get d_{KL}. For $\beta = 2$, the Euclidean distance is d_{Euc} is obtained.

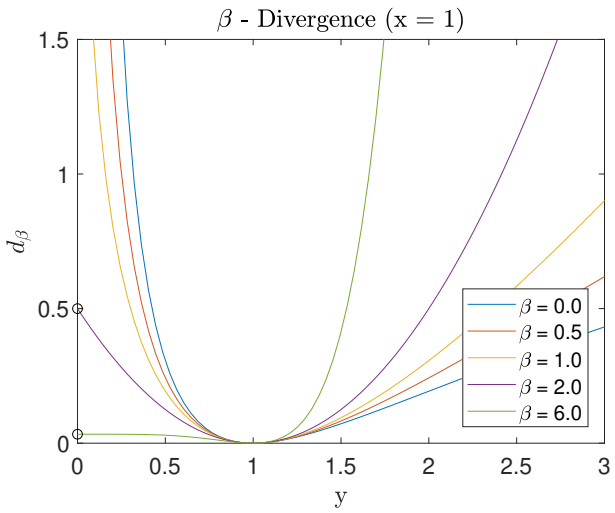


Figure 4.3 – $d_\beta(x, y)$ for different values of β. The divergences differ in the way they penalise too large or too small values of the approximation. Only for $\beta > 1$ the divergence takes finite values for $x = 0$.

4.1.2 Algorithms

In the optimization perspective of NMF, typically multiplicative update rules are used. These are based on the gradient descent algorithm, where during every iteration the coefficients are updated according to the following rule

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \eta_i \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}_i). \tag{4.10}$$

In [LS01] the well-known multiplicative update rules have been derived by a special choice of the step-size parameter η. Whereas η is typically a scalar value, it is extended to a matrix here. "$\circ$" denotes the element-wise product. Left of the arrow sign is the updated matrix $\boldsymbol{\theta}_{i+1}$. For a detailed derivation of the multiplicative update rule for the Euclidean distance, see appendix A.

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \boldsymbol{\eta} \circ \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}) \tag{4.11}$$

The main advantage of having formulated the divergences as a Bregman divergence is that the update equations do not have to be derived for each cost function individually, but all that is required to form a new multiplicative update rule is the second derivative of $\phi(x)$ and the formula derived in [SD06], which states

$$\mathbf{H} \leftarrow \mathbf{H} \circ \frac{\mathbf{W}^T(\nabla^2\phi(\mathbf{WH}) \circ \mathbf{X})}{\mathbf{W}^T(\nabla^2\phi(\mathbf{WH}) \circ \mathbf{WH})}, \tag{4.12}$$

$$\mathbf{W} \leftarrow \mathbf{W} \circ \frac{(\nabla^2\phi(\mathbf{WH}) \circ \mathbf{X})\mathbf{H}^T}{(\nabla^2\phi(\mathbf{WH}) \circ \mathbf{WH})\mathbf{H}^T}. \tag{4.13}$$

For the above choice ϕ_β, the second derivative is

$$\frac{d^2\phi_\beta(x)}{dx^2} = x^{\beta-2}, \tag{4.14}$$

so the multiplicative update equations for the general β divergence are [FBD09]

$$\mathbf{H} \leftarrow \mathbf{H} \circ \frac{\mathbf{W}^T((\mathbf{WH})^{\cdot(\beta-2)} \circ \mathbf{X})}{\mathbf{W}^T(\mathbf{WH})^{\cdot(\beta-1)}}, \tag{4.15}$$

$$\mathbf{W} \leftarrow \mathbf{W} \circ \frac{((\mathbf{WH})^{\cdot(\beta-2)} \circ \mathbf{X})\mathbf{H}^T}{(\mathbf{WH})^{\cdot(\beta-1)}\mathbf{H}^T}. \tag{4.16}$$

Thanks to these generalizations, the same implementation can easily be used for different values $\beta \in \mathcal{R}_+$, but choosing a optimal value is not a trivial problem. To do so, several approaches are imaginable. The tuning parameter β could either be found empirically as in [Coy09], where the energy based criteria (cf. chapter 7) are used as a quality measures, or it could be learned autonomously from a bigger data-set. An other approach is using statistical models to justify the decision, which is of particular interest in this work, cf. section 4.3.

4.1.3 Constraints

Within the optimization framework of NMF, additional properties of the two resulting matrices can be induced when extending the cost function by additional terms to obtain a constrained cost function D_C. The choice of the parameters α controls the effect of the constraints on the overall result.

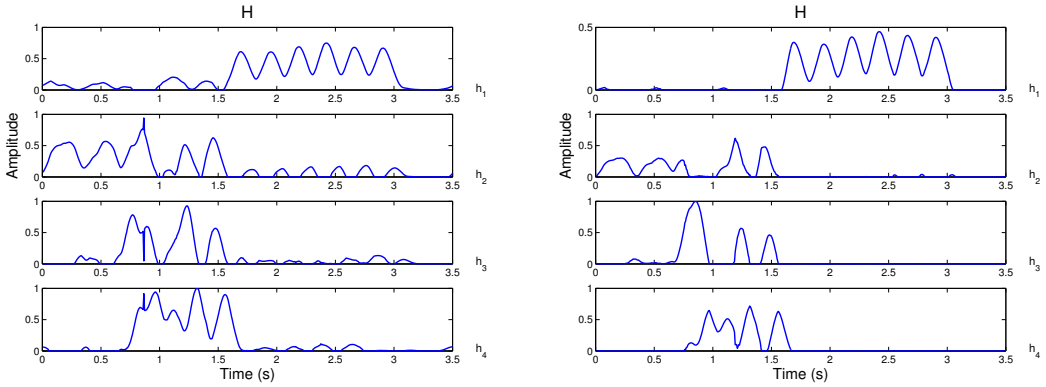
$$D_C(\mathbf{X}|\mathbf{WH}) = D(\mathbf{X}|\mathbf{WH}) + \alpha_W C_W(\mathbf{W}) + \alpha_H C_H(\mathbf{H}) \tag{4.17}$$

Sparse NMF. One possible goal is to enforce sparsity of either the columns of $\mathbf{W}$ or the rows of $\mathbf{H}$. This becomes increasingly important when using large numbers of components K, since the decomposition might yield a low divergence, but the interpretability of the components is lost. A simple sparsity criterion is the L_1-norm of $\mathbf{H}$. Since the elements of $\mathbf{H}$ are non-negative by constraint, it is simply the sum of the elements

$$C_H(\mathbf{H}) = \sum_k \sum_n h_{kn}. \tag{4.18}$$

In one of the earlier papers about sparse NFM [Hoy02], the corresponding update rule for the Euclidean distance is derived. When using the sparseness constraints on $\mathbf{H}$ it is important to carry out a normalization on $\mathbf{W}$, since otherwise the additional cost C_H is decreased easily with increasing values of $\mathbf{W}$.

One option is to normalize with the norm of the matrix $\mathbf{W}$. In [Hoy02], it is noted that the most meaningful way is normalizing the columns of $\mathbf{W}$, such that $||\mathbf{w}_k|| = 1$, but that the multiplicative update rules are no longer guaranteed to be non-increasing. The authors propose using additive gradient descent updates instead. In [Rou15], a pair of multiplicative update rules incorporating sparsity, column-wise normalization and general β-Divergences is derived, leading to an effective implementation of sparse NMF.



(a) (b)

Figure 4.4 – $\mathbf{H}$ from the NMF result of a short drum fill from "The End" by the Beatles. (a) without sparseness constraint. (b) with sparness constraint according to [Rou15], $\alpha_H = 0.5$. By penalizing large values of $\mathbf{H}$, the sparseness constraint produces less leakage between the components. The six peaks at the end correspond to six tom beats.

4.1.4 Reconstruction

Although some effort has been made to extend NMF to complex spectrograms, estimating both amplitude and phase at the same time, most algorithms are still based on the magnitude or power spectrograms. After having obtained the two matrices, time frequency representations of the separated components are calculated by multiplying each column $\mathbf{w}_k$ of $\mathbf{W}$ with each row $\mathbf{h}_k^T$ of $\mathbf{H}$. Typically these spectrograms are used as masks in the sense of Wiener filtering before the inverse frequency transformation. The complex spectrograms of the separated sources $\hat{\underline{\mathbf{S}}}_k$ become

$$\hat{\underline{\mathbf{S}}}_k = \frac{\mathbf{w}_k \mathbf{h}_k^T}{\mathbf{WH}} \circ \underline{\mathbf{X}} = \frac{\mathbf{Y}_k}{\mathbf{Y}} \circ \underline{\mathbf{X}}, \tag{4.19}$$

which means that effectively the phase of the original signal is used.

Griffin and Lim. During the inverse transform, the Griffin and Lim algorithm provides a way of recursively estimating a phase of the separated signals, which might provide a better fit to their magnitude [GL84]. Just as in a study comparing quality ratios for different phase recovery strategies [MBD16], no improvement in audio quality could be observed in case of the present separation results.

4.2 NTF

In non-negative tensor factorization, the information available in multichannel data is used explicitly. The spectrogram matrix $\mathbf{X} \in \mathbb{R}_+^{F \times N}$ is extended to form the spectrogram tensor $\mathbf{X} \in \mathbb{R}_+^{F \times N \times M}$. When performing the separation, a gain is assigned to every component for each channel. The gains are collected in the matrix $\mathbf{G} \in \mathbb{R}_+^{M \times K}$, where columns $\mathbf{g}_k$ describe the gains of the component k in the M channels. This is equivalent to assuming static sound sources, which is covered by the model from section 3.1. The reconstruction $\mathbf{Y}$ can be written as

$$\mathbf{X} \approx \mathbf{Y} = \sum_k \mathbf{g}_k \otimes \mathbf{w}_k \otimes \mathbf{h}_k^T, \tag{4.20}$$

where $\otimes$ denotes outer matrix multiplication. In index notation the same model reads

$$x_{nfm} \approx y_{nfm} = \sum_k g_{km} w_{fk} h_{kn}. \tag{4.21}$$

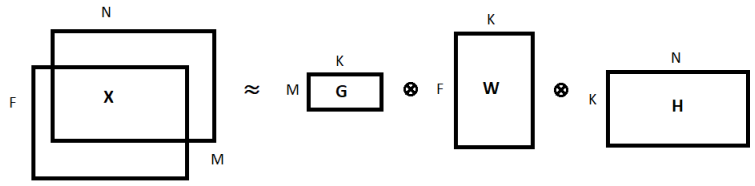


Figure 4.5 – Schematic representation of the matrices involved in NTF.

Tensor Products. It should be noted that in this context, the term "tensor" is simply meant to describe a multidimensional matrix. Apart from the outer product, which constructs the reconstructed spectrogram tensor, inner products are important for the algorithms. For 2D matrices, taking the inner product means multiplying and summing over the inner dimension of the two factors. The elements of the matrix $\mathbf{C} = \mathbf{AB}$ are computed by

$$c_{j_1 j_3} = \sum_{j_2} a_{j_1 j_2} b_{j_2 j_3}. \tag{4.22}$$

For tensors, the same inner product can be defined, executed over the inner-most dimension. Let $\mathbf{A}$ be a $J_1 \times ... \times J_k$ tensor and $\mathbf{B}$ be a $J_k \times ... \times J_K$ tensor, their inner product could be defined as

$$c_{j_1, j_{k-1}, j_{k+1}, j_K} = \sum_{j_k} a_{j_1, ..., j_k} b_{j_k, ..., j_K}. \tag{4.23}$$

It is even possible to choose several matching dimensions over which the product should be computed, no matter if they are the inner ones or not. This more general constracted tensor multiplication, will be denoted as in [FCC05]. For example, let $\mathbf{A}$ be a tensor with dimensions $I_1 \times ... \times I_N \times J_1 \times ... \times J_A$ and $\mathbf{B}$ a tensor with dimensions $I_1 \times ... \times I_N, L_1 \times ... \times L_B$. The product can be executed over all matching dimensions, which are indicated in curly brackets. In the following, the left set of dimensions corresponds to the left tensor and the right set to the right tensor. The product

$$\mathbf{C} = \langle \mathbf{AB} \rangle_{\{1,...,N,1,...,N\}}, \tag{4.24}$$

is equivalent to

$$c_{j_1,...,j_A,l_1,...,l_B} = \sum_{i_1} ... \sum_{i_N} a_{i_1,...,i_N,j_1,...,j_A} b_{i_1,...,i_N,l_1,...l_B}. \tag{4.25}$$

Algorithm. Multiplicative update rules for the tensor factorization with Kullback-Leibler divergence have been introduced in [FCC05]

$$\mathbf{G} = \mathbf{G} \circ \frac{\langle \mathbf{PD} \rangle_{\{1,2;2,3\}}}{\langle \mathbf{PO} \rangle_{\{1,2;2,3\}}}, \tag{4.26}$$

$$\mathbf{W} = \mathbf{W} \circ \frac{\langle \mathbf{QD} \rangle_{\{1,2;2,3\}}}{\langle \mathbf{QO} \rangle_{\{1,2;2,3\}}}, \tag{4.27}$$

$$\mathbf{H} = \mathbf{H} \circ \frac{\langle \mathbf{RD} \rangle_{\{1,2;1,2\}}}{\langle \mathbf{RO} \rangle_{\{1,2;1,2\}}}, \tag{4.28}$$

where the following auxiliary tensors have to be defined. $\mathbf{D} \in \mathbb{R}_+^{F \times N \times M}$ is the ratio of data and reconstruction

$$\mathbf{D} = \frac{\mathbf{X}}{\mathbf{Y}}. \tag{4.29}$$

The second one, $\mathbf{P} \in \mathbb{R}_+^{F \times N \times K}$ is the outer product of the component without considering the gains

$$\mathbf{P}_k = \mathbf{w}_k \mathbf{h}_k^T. \tag{4.30}$$

The other two auxiliary tensors $\mathbf{Q} \in \mathbb{R}_+^{M \times N \times K}$ and $\mathbf{R} \in \mathbb{R}_+^{M \times F \times K}$ are filled with

$$\mathbf{Q}_k = \mathbf{g}_k \mathbf{h}_k^T, \tag{4.31}$$

$$\mathbf{R}_k = \mathbf{g}_k \mathbf{w}_k^T. \tag{4.32}$$

These update rules where successfully tested on stereo and FOA data using an decoding/encoding step, cf. chapter 5. The NTF problem has also been brought to the statistical world, which will be explained in the next section.

4.3 Statistical Interpretation

Apart from the described optimization perspective of NMF, an alternative viewpoint has developed, embedding the NMF problem in a statistical framework, which is based on Bayes' law, maximum likelihood estimation, etc. Apart from a wider understanding of the problem, this approach bears several advantages, such as the simple incorporation of prior distributions, which motivate the constraints used in the previously described viewpoint. Also, this formulation inspires new NMF algorithms, stemming from the discipline of statistical signal processing.

4.3.1 Bayes' Law

This theorem, named after the English reverent *Thomas Bayes (1701-1761)*, is one of the most important foundations of statistical signal processing. It links probability density functions (pdfs) incorporating observation and prior knowledge in the following way

$$p(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{X})}$$

$p(\boldsymbol{\theta}|\mathbf{X})$ ... posterior
$p(\mathbf{X}|\boldsymbol{\theta})$ ... likelihood
$p(\boldsymbol{\theta})$ ... prior
$p(\mathbf{X})$ ... evidence

In case of the NMF problem Bayes' law gives an answer to the the following question for each time-frequency-bin (T-F bin): "*What is the probability of each entry in* $\mathbf{W}$ *and* $\mathbf{H}$ *to have a certain value, when having observed* $\mathbf{X}$*?*". Different initial beliefs about how $\mathbf{W}$ and $\mathbf{H}$ are distributed can be incorporated by the prior densities, which is one of the major advantages of this approach. In a static model, the evidence $p(\mathbf{X})$ acts as a normalization and needn't be considered in the estimation of $\mathbf{W}, \mathbf{H}$.

$$p(\mathbf{W}, \mathbf{H}|\mathbf{X}) \propto p(\mathbf{X}|\mathbf{W}, \mathbf{H})p(\mathbf{W})p(\mathbf{H}) \tag{4.33}$$

4.3.2 Maximum Likelihood Estimation

Maximum likelihood (ML) estimation doesn't exploit the full message of Bayes' law, but focusses on the likelihood function. For an ML estimator, we are looking to maximize the likelihood, or in practise minimize it's negative logarithm

$$(\mathbf{W}, \mathbf{H}) = \arg\min_{\mathbf{W},\mathbf{H}} -\log p(\mathbf{X}|\mathbf{W}, \mathbf{H}). \tag{4.34}$$

The likelihood tries to answer the question: *"How probable is a certain value of x_{fn} when having obtained y_{fn}?"*. What becomes explicit thereby is the fact that the factorization is only an approximation of $\mathbf{X}$, so the value x_{fn} doesn't correspond to y_{fn} exactly, but is somehow distributed in dependence of it. There are many different options for guessing how this distribution might be, but some are more popular then others, mainly for one reason: The resulting ML estimator corresponds exactly to one of the well known cost functions described above.

Gaussian Likelihood model I. In [FBD09] the IS divergence is derived from a Gaussian likelihood using the following model. The complex samples of the individual sources' spectrograms are assumed to be distributed according to a complex Gaussian distribution, which is centered around zero, and is scaled by the samples of the estimated sources

$$p(\underline{s}_{fnk}) = \mathcal{N}_c(0, w_{fk}h_{kn}). \tag{4.35}$$

In case of a zero mean and a uniformly distributed phase, the complex normal distribution is called circularly-symmetric and it's pdf is given by

$$\mathcal{N}_c(\underline{x}|0, \sigma^2) = \frac{1}{\pi\sigma^2} e^{-\frac{|\underline{x}|^2}{\sigma^2}}. \tag{4.36}$$

The samples of the mixed signal's spectrogram are the sum of the source spectrograms' samples

$$\underline{x}_{fn} = \sum_k \underline{s}_{fnk}. \tag{4.37}$$

The pdf of the sum of two random variables is determined by the convolution of the variables' pdfs. Despite the fact that complex distributions actually describe two values, real and imaginary part, the circularly-symmetric complex Gaussian distribution depends on the norm only. This means that it is actually 1-dimensional and the property of the normal distribution, which states that the sum of normally distributed variables is normally distributed [ES08], can be translated to this case as well. The pdf of the summed circularly symmetric Gaussian is a circularly symmetric Gaussian itself, where the variance is the sum of the variances and the likelihood is described by

$$p(\underline{x}_{fn}|w_{f:}h_{:n}) = \frac{1}{\pi \sum_k w_{fk}h_{kn}} e^{-\frac{|\underline{x}_{fn}|^2}{\sum_k w_{fk}h_{kn}}} = \frac{1}{\pi y_{fn}} e^{-\frac{|\underline{x}_{fn}|^2}{y_{fn}}}. \tag{4.38}$$

Summed over all T-F bins, the negative log likelihood can be written as

$$D_{ML}^{\mathcal{N}} = -\sum_n \sum_f \log \frac{1}{\pi y_{fn}} e^{-\frac{|\underline{x}_{fn}|^2}{y_{fn}}} \tag{4.39}$$

$$= \sum_n \sum_f (\log\pi + \log y_{fn} + \log e^{-\frac{|\underline{x}_{fn}|^2}{y_{fn}}}), \tag{4.40}$$

$$= NF\log\pi + \sum_n \sum_f (\log y_{fn} + \frac{|\underline{x}_{fn}|^2}{y_{fn}}). \tag{4.41}$$

The resulting expression is equivalent to the IS divergence D_{IS} up to a constant, when the power spectrogram is used as an input. This means that they have the same maximal point.

Poisson Likelihood Model. In the same way, other distributions can be associated with the different divergences. In [Cem09] the Poisson distribution is chosen to formulate the source model. According to this model, the magnitudes of the source samples $s_{fn} = |\underline{s}_{fkn}|$ are distributed with

$$s_{fnk} \sim \mathcal{PO}(w_{fk}h_{kn}), \tag{4.42}$$

where $\mathcal{PO}$ is the Poisson distribution, defined by

$$\mathcal{PO}(x;\lambda) = \frac{e^{-\lambda}\lambda^x}{\Gamma(x+1)}, \tag{4.43}$$

and $\Gamma(x)$ is the gamma function. Note that the Poisson distribution is formally defined for integers only. This does not seem to correspond to reality, but the model can still be used, since the spectrogram can be arbitrarily scaled. Furthermore, the Poisson model doesn't represent a sum of complex samples or leads to the power spectrogram, but is defined for the magnitude of the samples. The modelling assumption states that the magnitude spectrogram is the sum of the source magnitudes, which is not physical. The full derivation can be found in [VCG08].

$$x_{fn} = \sum_k s_{fnk} \tag{4.44}$$

To derive the ML estimator, we can use the rule which states that the sum of independent Poisson distributed random variables is also Poisson, where the new intensity parameter is the sum of the old ones. This leads to the likelihood

$$p(x_{fn}|\mathbf{w}_f, \mathbf{h}_n) = \mathcal{PO}(x_{fn}|\sum_k w_{fk}h_{kn}). \tag{4.45}$$

The negative log-likelihood is

$$D_{ML}^{\mathcal{PO}} = -\log \prod_f \prod_n \frac{e^{-y_{fn}} y_{fn}^{x_{fn}}}{\Gamma(x_{fn}+1)} \tag{4.46}$$

$$= \sum_f \sum_n -y_{fn} + x_{fn}\log(y_{fn}) - \log(\Gamma(x_{fn}+1)). \tag{4.47}$$

This is equivalent to the Kullback-Leibler divergence D_{KL}, used by many researchers in NMF.

Tweedie Likelihood Model. It was even noticed, that there is a family of distributions, based on the Tweedie distribution that, when used a likelihood, would result in the family of β-divergences defined above [TF13].

4.3.3 Maximum A Posteriori Estimation

One step further into the statistical framework lies the maximum a posteriori (MAP) estimator. Instead of focusing on the likelihood alone, the MAP takes prior densities of $\mathbf{W}$ and $\mathbf{H}$ into account and maximises the posterior

$$(\mathbf{W}, \mathbf{H}) = \arg\min_{\mathbf{W},\mathbf{H}} -\log p(\mathbf{W}, \mathbf{H}|\mathbf{X}). \tag{4.48}$$

Prior densities take the place of the constraints in the optimization framework.

Markov Chain Priors. A common choice is to use a prior on $\mathbf{H}$, which enforces smoothness. This can be done by the following prior structure

$$p(h_{kn}) = \prod_{n=1}^{N-1} p(h_{kn}|h_{k(n-1)}) p(h_{k,0}), \tag{4.49}$$

where $p(h_{kn}|h_{k(n-1)})$ is a pdf with it's mode at $h_{k(n-1)}$.

This definition is given in [FBD09], where a Gamma chain is used and the update is carried out using space-alternating generalized expectation-maximization (SAGE). In [VTG08], a Gamma chain is applied as well, introducing an auxiliary variable z_{kn}. Increasing values of α_H result in the a stronger coupling of the $\mathbf{H}$ entries over time

$$p(z_{k,0}) = \mathcal{G}(z_{k,0}; \alpha_H + 1, \alpha_H \beta_H), \tag{4.50}$$
$$p(h_{kn}|z_{kn}) = \mathcal{G}(h_{kn}|\alpha_H, \alpha_H z_{kn}), \tag{4.51}$$
$$p(z_{k(n+1)}|h_{kn}) = \mathcal{G}(z_{k(n+1)}|\alpha_H + 1, \alpha_H h_{kn}). \tag{4.52}$$

To solve this, multiplicative update rules are used. Similar to the additive constants which follow from the simple constraints shown above, additive terms appear in the numerator and denominator of the update rule for $\mathbf{H}$. In this framework, this update can not be done matrix-wise any more

$$w_{fk} \leftarrow w_{fk} \frac{\sum_n h_{kn} \frac{x_{fn}}{y_{fn}}}{\sum_n h_{kn}}, \tag{4.53}$$
$$h_{kn} \leftarrow h_{kn} \frac{\frac{2\alpha_H}{h_{kn}} + \sum_n h_{kn} \frac{x_{fn}}{y_{fn}}}{\alpha_H (z_{kn} + z_{k(n+1)}) + \sum_n h_{kn}}, \tag{4.54}$$
$$\tag{4.55}$$
$$z_{kn} \leftarrow \begin{cases} \frac{1}{h_{k,0} + \beta_H} & n = 0 \\ \frac{2}{h_{kn} + h_{k(n-1)}} & n = 1, ..., N-1 \\ \frac{1}{h_{kn}} & n = N \end{cases}. \tag{4.56}$$

Note that apart from the introduced additive terms, the updates are equivalent to the Kullback-Leibler (or β-Divergence with $\beta = 1$). This is because a Poisson likelihood is used as shown above.

4.4 Full Bayesian Inference

The derivation of the ML and MAP estimator lead to a new understanding of the cost functions, but they are still point estimators, whose solution is typically found by numerical optimization. Furthermore, it has been shown, that the two statistical models presented above yield an ML estimator which is equivalent to the known divergences. This means that even exactly the same algorithms can be used as before. This is different when applying a full Bayesian treatment, where not only a maximal point of the posterior distribution will be found, but the entire distribution will be approximated by actually generating samples from it.

4.4.1 Gaussian Likelihood Model II

In [SWH09] and [BFL17], this process is described, based on a Gaussian model, which differs from the one described above. Here, the true values of the magnitude spectrogram $\mathbf{X}$ are assumed to be distributed around the reconstructed values $\mathbf{Y}$ with unknown variance σ^2. This model might seem more intuitive than the model used in the IS divergence derivation in [FBD09], which has zero mean and the variance scaled by the value of the approximation. Figure 4.6 shows a schematical comparison, where an arbitrary value of x_{fn} is kept constant, while the reconstruction value is scaled with a constant.

$$x_{fn} \sim \mathcal{N}(y_{fn}, \sigma^2). \tag{4.57}$$

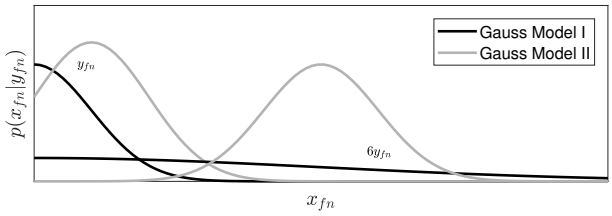


Figure 4.6 – Schematical comparison of Gaussian Likelihood Model I and Model II for a certain value y_{nf} and a multiple of it ($6y_{nf}$).

The unknown variance is distributed according to the inverse gamma distribution

$$\sigma^2 \sim \mathcal{G}^{-1}(\alpha, \beta). \tag{4.58}$$

The inverse gamma distribution is the conjugate prior for a random variance of the normal distribution. The posterior of a variable with a conjugate prior belongs to the same family of distributions as the posterior again. In this case it is even exactly the inverse gamma distribution again (cf. A.44), which is defined as

$$\mathcal{G}^{-1}(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}(x)^{-\alpha-1}e^{-\frac{\beta}{x}}. \tag{4.59}$$

Furthermore, exponential priors are introduced for the elements of $\mathbf{W}$ and $\mathbf{H}$. The rate parameter of these exponentials has a direct influence on the separation's sparsity.

$$w_{fk} \sim \mathcal{E}(\lambda_{fk}^{W}) \qquad h_{kn} \sim \mathcal{E}(\lambda_{kn}^{H}) \tag{4.60}$$

At the same time as the exponential priors resemble the sparse distribution of samples in the dictionary and the activation matrices, it causes non-negativity of the entries. No explicit non-negativity constraint is required. The exponential distributions' density is defined as

$$\mathcal{E}(x;\lambda) = \lambda e^{-\lambda x} u(x), \tag{4.61}$$

where $u(x)$ is the Heaviside step function.

4.4.2 Gibbs Sampling

With the model at hand, a Gibbs sampler for drawing samples from the conditional posterior densities can be derived. The Gibbs sampler belongs to the class of Markov Chain Monte Carlo (MCMC) methods.

The important idea behind the Gibbs Sampler is that when sequentially sampling from the conditional posteriors of all parameters, the draws will converge to draws of the joint posterior. Thereby, sequential sampling from a very high-dimensional model becomes possible. For this, the posterior of every variable, depending on all other variables needs to be derived. In each iteration of the sampler, samples will be drawn from these distributions

$$p(\theta_j^{(i+1)} | \theta_1^{(i+1)}, \theta_2^{(i+1)}, ..., \theta_{j-1}^{(i+1)}, \theta_{j+1}^{(i)}, ..., \theta_J^{(i)}). \tag{4.62}$$

After a few iterations, the sampler is said to have "burned-in". Then, samples from the joint posterior are produced. Due to the concept of MCMC, the samples following each other are not fully uncorrelated. To cope with this, thinning can be applied by only taking samples from every e.g. third iteration.

Conditonal Mean. To use the result of the sampling procedure, ultimately one value has to be determined for every variable, on which the separation is based. One way of doing this is to apply the conditional mean estimator. For this, the mean over all realisations after burn-in I_{burn} and possible thinning is computed

$$\theta_j^{CM} = \frac{1}{I} \sum_{i=1}^{I} \theta_j^{(i)}. \tag{4.63}$$

Gibbs Sampler for Gaussian Model II. Using Bayes' law, the conditional posterior for one single sample w_{fn} of the first factor matrix $\mathbf{W}$ can be derived. Every draw depends on all other parameters, except the specific w_{fn}. The derivation required to obtain the posterior distributions is executed in A.3.1 in much detail. The resulting posterior is a truncated Gaussian

$$p(w_{fk}|\mathbf{X}, \mathbf{W}_{\setminus(w_{fk})}, \mathbf{H}, \sigma^2) = p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \sigma^2)p(w_{fk}; \lambda^W_{fk}) \tag{4.64}$$

$$= \prod_n \mathcal{N}(\mathbf{X}|\mathbf{WH}, \sigma^2)\mathcal{E}(w_{fk}; \lambda^W_{fk})u(w_{fk}) \tag{4.65}$$

$$= \mathcal{TN}(w_{fk}|\mu_W, \sigma^2_W)_{[0,\infty)}, \tag{4.66}$$

with mean and variance

$$\sigma^2_W = \frac{\sigma^2}{\sum_n h^2_{kn}} \tag{4.67}$$

$$\mu_W = \sigma^2_W\big(\frac{1}{\sigma^2}\sum_n(x_{fn}h_{kn} - h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda^W_{fk}\big). \tag{4.68}$$

When looking at the rate λ^W_{fk} of the exponential factor prior, it becomes clear how the prior affects the result. It pulls the mean further towards the negative direction, thereby making smaller values more probable if the rate of the exponential prior is chosen to be high. λ^W_{fk} can be chosen to be the same for every entry, or it could be low in a harmonic grid and high elsewhere, encouraging certain notes.

The posterior for the elements of factor $\mathbf{H}$ is found, completely analogous, in a truncated Gaussian with the parameters

$$\sigma^2_H = \frac{\sigma^2}{\sum_f w^2_{fk}} \tag{4.69}$$

$$\mu_H = \sigma^2_H\big(\frac{1}{\sigma^2}\sum_f(x_{fn}w_{fk} - w_{fk}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda^H_{kn}\big). \tag{4.70}$$

As mentioned before, since the inverse gamma distribution is a conjugate prior for the Gaussian Likelihood with unknown variance, the posterior density of the variance is inverse gamma as well. The derivation yields the following parameters

$$\alpha' = \frac{NF}{2} + \alpha \tag{4.71}$$

$$\beta' = \frac{1}{2}\sum_f\sum_n(x_{fn} - \sum_k w_{fk}h_{kn})^2 + \beta. \tag{4.72}$$

Also this result is easily interpretable. β' decreases with the fit of data and reconstruction, making the variance of the drawn samples smaller. Since the inverse gamma distribution is only non-zero away from $\sigma^2 = 0$, the sampler will never completely freeze, which would force it to create the same values over and over again.

Computation. The samples from $\mathbf{W}$ can be drawn column-wise and the samples of $\mathbf{H}$ row-wise, component for component. As described in [SWH09] already, the computational effort can be reduced by pre-computing some matrix products before drawing the component samples. Taking for example the posterior draw of $\mathbf{W}$, the product $\sum_n x_{fn} h_{kn}$ can be computed as one matrix product $\mathbf{X}\mathbf{H}^T$ before sampling. The same holds for the product $\sum_f x_{fn} w_{fk}$, which is computed as $\mathbf{W}^T\mathbf{X}$ before sampling from $\mathbf{H}$. The complete algorithm can be written in the following way. Note that the values $\lambda^W_{fk}, \lambda^H_{kn}$ are collected in the matrices $\mathbf{\Lambda}^W, \mathbf{\Lambda}^H$.

Data: $\mathbf{X}$, σ^2, $\mathbf{\Lambda}^W$, $\mathbf{\Lambda}^H$, α_0, β_0
Result: $\{\mathbf{W}^{(i)}, \mathbf{H}^{(i)}\}_{i=1}^{I}$
$\mathbf{W} \leftarrow \mathcal{U}(0,1)$;
$\mathbf{H} \leftarrow \mathcal{U}(0,1)$;
$\alpha \leftarrow \alpha_0 + \frac{FN}{2}$;
for *i to I* **do**
  $\mathbf{A} \leftarrow \mathbf{X}\mathbf{H}^T$;
  $\mathbf{V}^H \leftarrow \mathbf{H}\mathbf{H}^T$;
  for *k < K* **do**
    $\sigma^2_W \leftarrow \frac{\sigma^2}{v^H_{kk}}$;
    $\boldsymbol{\mu}^2_W \leftarrow \sigma^2_W(\frac{1}{\sigma^2}(\mathbf{A}_{:,k} - \mathbf{W}_{:,\setminus k}\mathbf{V}^H_{\setminus k,k}) - \mathbf{\Lambda}^W_{:,k})$;
    $\mathbf{W}^{(i)}_{:,k} \leftarrow \mathcal{TN}(\boldsymbol{\mu}_W, \sigma^2_W)$;
  end
  $\beta \leftarrow \beta_0 + \sum_{fn}(\mathbf{X} - \mathbf{W}\mathbf{H})^2$;
  $\sigma^2 \leftarrow \mathcal{G}^{-1}(\alpha, \beta)$;
  $\mathbf{B} \leftarrow \mathbf{W}^T\mathbf{X}$;
  $\mathbf{V}^W \leftarrow \mathbf{W}^T\mathbf{W}$;
  for *k < K* **do**
    $\sigma^2_H \leftarrow \frac{\sigma^2}{v^W_{kk}}$;
    $\boldsymbol{\mu}^2_H \leftarrow \sigma^2_H(\frac{1}{\sigma^2}(\mathbf{B}_{k,:} - \mathbf{V}^W_{k,\setminus k}\mathbf{H}_{\setminus k,k} - \mathbf{\Lambda}^H_{k,:})$;
    $\mathbf{H}^{(i)}_{k,:} \leftarrow \mathcal{TN}(\boldsymbol{\mu}_H, \sigma^2_H)$
  end
end

Algorithm 1: NMF Gibbs sampler.

4.4.3 Sampling from the Truncated Gaussian

Naive Accept/Reject Method. Given an algorithm which samples from the Gaussian, the simplest sampling method is rejection sampling. Drawing samples from a Gaussian is implemented in numerous software packages. In Matlab, the Ziggurat algorithm is used [Mol08]. To obtain samples from the Gaussian truncated on $[a, b]$, one could have the idea to draw samples from $\mathcal{N}$, until a value within the truncation interval appears. If N non-negative samples on the interval $[0, \infty)$ should be drawn with $\mu = 0$, the rejection sampler will on average need to sample $2N$ times, but the exact number of required runs can not be determined beforehand. A problem occurs, if the mean lies far out of the truncation interval (eg. $\mu \ll 0$ for $[0, \infty)$) this rejection method will need extremly large numbers of runs, which is highly impractical [Rob95].

Inverse Transform Method. First of all, it should be noticed that it is sufficient to implement a sampler, which is capable of drawing samples from a normal distribution with zero mean and unit variance truncated on $[a, b]$. For other parameters, one may scale the limits and the result accordingly

$$x \sim \mathcal{N}(\mu, \sigma^2)_{[a,b]} \tag{4.73}$$

$$x = \mu + \sigma z \quad z \sim \mathcal{N}(0, 1)_{[(\frac{a-\mu}{\sigma}),(\frac{b-\mu}{\sigma})]}. \tag{4.74}$$

The inverse transform method is a straight forward algorithm for sampling from a distribution with the inverse cdf $\Phi^{-1}(x)$ using a sampler that creates uniformly distributed data $u \sim \mathcal{U}[0, 1]$. The procedure is based on computing $\Phi^{-1}(u)$. To match the truncation interval, the argument needs to be scaled such that only values between $\Phi(a)$ and $\Phi(b)$ are fed to the inverse cdf. For a standard truncated normal, the sampler is given by

$$x \sim \Phi^{-1}(\Phi(a) + u(\Phi(b) - \Phi(a))), \tag{4.75}$$

where $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}t^2} dt$ is the Gaussian cdf. Although no multiple draws are required with this method, it is likely to fail if the mean is to far outside the truncation interval, due to the numerical accuracy of the cdf. A small simulation demonstrates this.

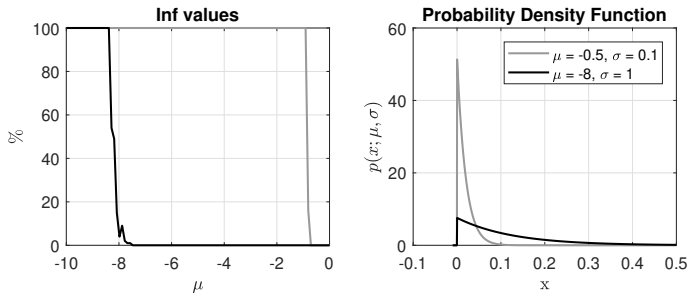


Figure 4.7 – (Left) Proportion of Inf samples when using the inverse transform method for the normal distribution truncated on $[0, \infty)$ and $\sigma^2 = [0.1, 1]$. (Right) the pdfs with the variances used and the means at which sampling fails. Carried out in Matlab 2017b.

Sampling the tail. Sampling from the far end of the distributions' tail occurs in cases where the mean is far away from the boundary and the variance is small. Apparently a different sampler has to be used in this scenario.

One option is to apply an accept/reject algorithm, based on the Rayleigh distribution. A proper reject/accept algorithm works by approximating the target pdf $p(x)$ by a propositional pdf $q(x)$, which fulfils $Mq(x) < p(x)$ for all non-zero values of $q(x)$ and some constant M. The proposal densities' support must include the support of $p(x)$. For the tail of the normal distribution, the Rayleigh distribution with unit scale fulfils these condition. Sampling from the Rayleigh distribution is done again by the inverse transform method using $u \sim \mathcal{U}(0,1)$

$$\Phi(x) = 1 - e^{\frac{x^2}{2}} \tag{4.76}$$

$$\Phi^{-1}(u) = \sqrt{-2\log(1-u)}. \tag{4.77}$$

Just as above, the inverse transform needs to be bound to the truncation interval. The resulting cdf can be expressed analytically by

$$x \sim \Phi^{-1}(\Phi(a) + u(\Phi(b) - \Phi(a))) \tag{4.78}$$

$$= \sqrt{-2\log(e^{\frac{-a^2}{2}} - u(e^{\frac{-a^2}{2}} - e^{\frac{-b^2}{2}}))} \tag{4.79}$$

$$= \sqrt{-2\log(e^{\frac{-a^2}{2}}(1 + u(e^{\frac{a^2-b^2}{2}} - 1)))} \tag{4.80}$$

$$= \sqrt{a^2 - 2\log(1 + u(e^{\frac{a^2-b^2}{2}} - 1))}. \tag{4.81}$$

For convenience, sampling can be done from the random variable $\frac{x^2}{2}$ instead, leaving square-root and multiplying by 2 for final result, Algorithm 4 from [BL17] is obtained.

The constant M is determined from the truncated Rayleigh density and the truncated normal density. The acceptance condition of the algorithm follows from this result.

Data: a, b
Result: $x \sim \mathcal{TN}(a,b)$
$c \leftarrow \frac{a^2}{2}$;
$q \leftarrow 1 - e^{c-\frac{b^2}{2}}$;
do
  $u \sim \mathcal{U}(0,1)$;
  $v \sim \mathcal{U}(0,1)$;
  $x \leftarrow c - \ln(1-qu)$
while $v^2 x \leq a$;
$x \leftarrow \sqrt{2x}$;

Algorithm 2: Sampler for the tail of $\mathcal{TN}_{[a,b]}$ [BL17].

4.4.4 Automatic Relevance Determination

In classical NMF, the selection of the model order (the number of components K) can be a very difficult problem. One option for model order selection is to run the separation with different orders and then select the result with the smallest divergence. Trying different orders is obviously very expensive in terms of calculation time and also the lowest divergence alone might not yield the most meaningful separation. Here, the statistical framework proves to be useful, as it makes it easy to incorporate an automatic relevance determination (ARD) scheme [TF13].

Modified Prior Structure. For automatic relevance determination (ARD), the existing model is slightly modified. Instead of an individual rate parameter of the exponential priors on each element of $\mathbf{W}$ and $\mathbf{H}$, only one rate λ_k is determined for each component, which is applied in both priors. This means that the prior structure is changed to

$$\mathbf{w}_k \sim \mathcal{E}(\boldsymbol{\lambda}_k^W), \qquad \mathbf{h}_k^T \sim \mathcal{E}(\boldsymbol{\lambda}_k^H), \tag{4.82}$$

where $\boldsymbol{\lambda}_k^W$ and $\boldsymbol{\lambda}_k^H$ are vector of dimensions $(F \times 1)$ and $(1 \times N)$ with λ_k at all entries. λ is no longer a user defined parameter, but a random variable. On this random variable, a gamma prior is placed.

$$\lambda_k \sim \mathcal{G}(\alpha_k^{(\lambda)}, \beta_k^{(\lambda)}) \tag{4.83}$$

The posterior for this gamma prior is a gamma distribution itself [BFL17]

$$p(\boldsymbol{\lambda}_k|\mathbf{X}, \mathbf{W}, \mathbf{H}, \sigma^2) = \mathcal{G}(\boldsymbol{\lambda}_k|\alpha_k^{(\lambda)}, \beta_k^{(\lambda)}), \tag{4.84}$$

which is parametrised by

$$\alpha_k^{(\lambda)} = \alpha_0^{(\lambda)} + F + N \tag{4.85}$$

$$\beta_k^{(\lambda)} = \beta_0^{(\lambda)} + \sum_f w_{fk} + \sum_n h_{kn}. \tag{4.86}$$

In Figure 4.9 a separation with a slightly overdetermined $K = 15$ on a short excerpt with bass and violin shows the functionality.

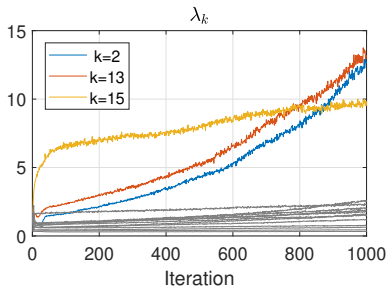


Figure 4.8 – λ_k during 1000 iterations of the algorithm. Three components exhibit rising λ values, the means of the samplers are thereby drawn toward the negative direction.

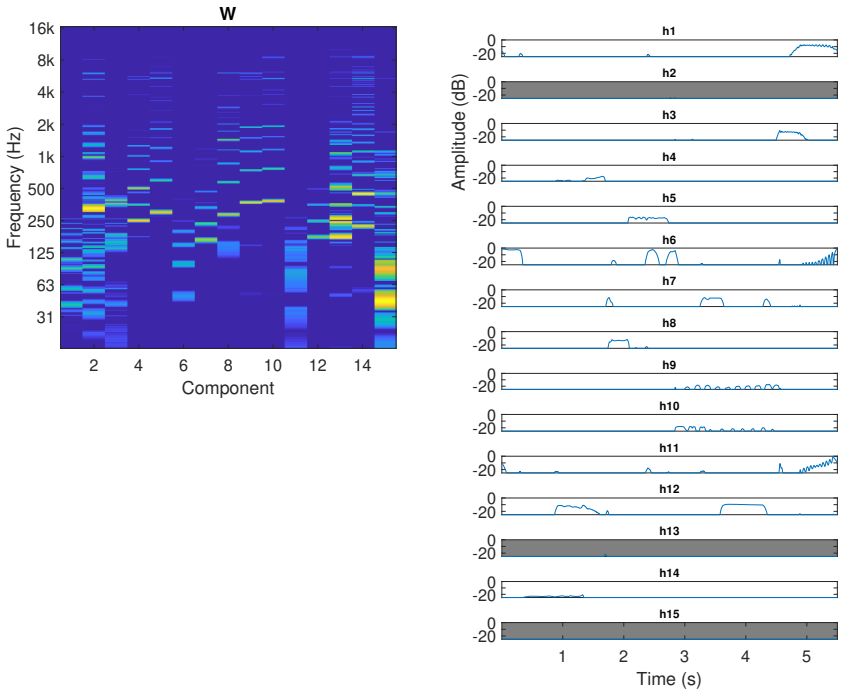


Figure 4.9 – Result of NMF separation with automatic relevance determination. The components, which exhibit high values for λ are "switched off" by the algorithm (highlighted in gray). $K = 15$, $N = 3580$, $F = 361$, $I = 1000$, prior parameters set to 0. Normalization only carried out with respect to the largest value in $\mathbf{W}$ and $\mathbf{H}$, 30 dB dynamics shown for $\mathbf{W}$.

4.4.5 Bayesian NTF

The same Gibbs sampling approach can be applied to the NTF Problem. The statistical model is extended by the gain factor matrix $\mathbf{G}$, for which a meaningful prior density needs to be found.

Uniform Gain Prior. After first experiments with exponential priors, for which all posterior densities can be derived analogously, the most successful results were found using a uniform prior over the gains

$$g_{km} \sim \mathcal{U}(0, 1). \tag{4.87}$$

The posterior is a Gaussian again, but this time truncated on the interval $[0, 1]$

$$\begin{aligned} p(g_{km}|\mathbf{X}, \mathbf{W}, \mathbf{H}, \mathbf{G}_{\setminus g_{km}}, \sigma^2) &= p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \mathbf{G}, \sigma^2)p(g_{km}) && (4.88)\\ &\propto \prod_f \prod_n \mathcal{N}(x_{fnm}|\sum_k w_{fk}h_{kn}g_{km}, \sigma^2)\mathcal{U}(g_{km}) && (4.89)\\ &= \mathcal{TN}(g_{km}|\mu_G, \sigma_G^2)_{[0,1]}, && (4.90) \end{aligned}$$

with mean and variance equal to

$$\sigma_G^2 = \frac{\sigma^2}{\sum_f \sum_n w_{fk}^2 h_{kn}^2} \tag{4.91}$$

$$\mu_G = \frac{\sigma_G^2}{\sigma^2} \sum_f \sum_n (x_{fnm} w_{fk} h_{kn} - w_{fk} h_{kn} \sum_{k' \neq k} w_{fk'} h_{k'n} g_{k'm}). \tag{4.92}$$

The modified parameters for the other factor posteriors are again derived in the appendix. With this sampler at hand, non-negative tensor factorisations can be computed where the the entries of the gain matrix $\mathbf{G}$ are non-negative. This could either be a stereo file, or a decoded Ambisonics signal, cf. chapter 5.

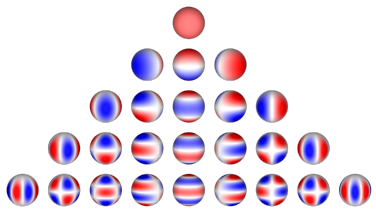
Chapter 5

A Brief Summary of Ambisonics

Ambisonics is a technique for recording, editing, storing and reproducing a surrounding soundfield. Since comprehensive literature is available [ZF19], the topic is not covered in detail, but the most important concepts are summarized, especially for understanding why the decoding/encoding step in NTF with ambisonic input is necessary and how one may apply a beamformer when processing the NTF results.

Spherical Harmonic Representation. The important concept behind Ambisonics is the description of the soundfield in terms of spherical harmonics. Opposed to transmitting a signal for a fixed amount of loudspeakers (discrete format) or a signal for every source along with metadata (object-based format), one signal is transmitted for every spherical harmonic channel. This could be referred to a scene-based format. Like this, the number of transmitted channels does not depend on the number of speakers used for playback, nor the number of sources encoded, but on the desired spatial resolution of the scene, which is determined by the Ambisonics order N_{sh}. In case of a 3D representation, the number of required channels is $(N_{sh}+1)^2$.

$$Y_n^m(\boldsymbol{\theta}) = \sqrt{\tfrac{(2n+1)(n-m)!}{4\pi(n+m)!}} P_n^m(\cos\vartheta) \begin{cases} \sin(|m|\varphi) & m<0 \\ 1 & m=0 \\ \cos(m\varphi) & m>0 \end{cases} \tag{5.1}$$



Encoding. To encode a sound source to the Ambisonics scene, each sample of the signal is simply multiplied by the spherical harmonics, evaluated at the desired panning direction. Using Ambisonics channel numbering (ACN, [NZDS11]) to determine the index, they can be stacked into a vector $\boldsymbol{y}$. A scene with K encoded sources at the directions $\boldsymbol{\theta}_k$ is described by

$$\boldsymbol{\chi}[t] = \sum_k \boldsymbol{y}(\boldsymbol{\theta}_k) s_k[t]. \tag{5.2}$$

First-Order Ambisonics. When only considering the zeroth and the first order harmonics ($N_{sh} = 1$), the description is rather simple. Most microphone technology is based on the first order, where the signals of four capsules in a tetrahedral arrangement are multiplied by an encoder matrix to form the Ambisonics signal.

$$\boldsymbol{\chi}[t] = \boldsymbol{s}[t]\frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{5.3}$$

Figure 5.1 – Soundfield ST450 FOA microphone and corresponding encoding matrix.

Decoding. To obtain signals for L loudspeakers from an Ambisonics signal, a decoding matrix needs to be applied

$$\boldsymbol{x}[t] = \boldsymbol{\chi}[t]\boldsymbol{D}. \tag{5.4}$$

The design of the decoding matrix has been an important problem in the development of Ambisonics. A successful way to solve this problem for almost arbitrary loudspeaker layouts is available with the AllRAD approach [ZF12]. For the special case of regularly spaced layouts on the sphere, which is usually not seen in practise, decoder design is simplified immensely. Such regularly spaced grids are called t-design. Perfect t-design layouts on the sphere only exist for $L \leq 20$. Above this limit, they can only be approximated. A tetrahedron is a t-design with $L = 4$, for which the properly scaled decoding matrix is

$$\boldsymbol{D} = \sqrt{\pi}\boldsymbol{Y}(\boldsymbol{\Theta}_L)^T = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{5.5}$$

Since the layout is perfectly regular, the matrix is orthogonal and the condition $\boldsymbol{D}^T\boldsymbol{D} = \boldsymbol{I}$ holds, which makes it possible to decode and re-encode a first order Ambisonics signal without loosing any information. For the tetrahedron, the matrix even is symmetric and thereby equal for encoding and decoding.

Weighting and "Non-Negative Ambisonics". Weighting influences the way the segment of the scene assigned to each loudspeaker is shaped. If no weighting is used, there is a relatively large amount of opposite side signal present in each speaker. If in-phase weights are used, there is no opposite information at all, but the width of the considered area around the speaker is large.

Max-r_E weighting offers a good compromise, where the opposite side is reduced at a lesser increase of the width. Before decoding, a weight is multiplied with every SH channel

$$\boldsymbol{x}[t] = \boldsymbol{\chi}[t]\text{diag}(\boldsymbol{a})\boldsymbol{D}. \tag{5.6}$$

For higher orders, max-r_E weights can be approximated by [ZF12]

$$a_n = \mathcal{P}_n\Big(\cos\frac{137.9^\circ}{N+1.51}\Big), \tag{5.7}$$

and in-phase weights are computed by [Dan00]

$$a_n = \frac{N!}{(N-n)!}\frac{(N+D+2)!}{(N+n+D-2)!}. \tag{5.8}$$

For the first order, the in-phase weights are $[1\ \frac{1}{3}\ \frac{1}{3}\ \frac{1}{3}]^T$ the max-r_E weights are $[1\ \frac{1}{\sqrt{3}}\ \frac{1}{\sqrt{3}}\ \frac{1}{\sqrt{3}}]^T$.

For the present application, in-phase weighting has a very useful property. In case of an encoded source at an arbitrary direction, the speaker weights after decoding with in-phase weighting are all positive (hence the name of the approach). This transformation is important for the NTF algorithm with ambisonic input, since no negative gains can be allowed for during separation.

$$\boldsymbol{x}[t] = \boldsymbol{\chi}[t]\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \frac{1}{3} & 0 & 0\\ 0 & 0 & \frac{1}{3} & 0\\ 0 & 0 & 0 & \frac{1}{3}\end{bmatrix}\boldsymbol{D} \tag{5.9}$$

After the NTF algorithm, the re-encoding operation is applied to the gain matrix, after which we obtain spherical harmonic weights for each component

$$\tilde{\mathbf{G}} = \sqrt{\pi}\mathbf{G}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \frac{1}{3} & 0 & 0\\ 0 & 0 & \frac{1}{3} & 0\\ 0 & 0 & 0 & \frac{1}{3}\end{bmatrix}^{-1}\boldsymbol{Y}(\boldsymbol{\Theta}_L). \tag{5.10}$$

Pseudo-Intensity Vector. Using the Pseudo-Intensity vector, the panning direction corresponding to these weights can be obtained

$$\boldsymbol{I}_k = \tilde{g}_{1,k}\cdot\begin{bmatrix}\tilde{g}_{4,k}\\ \tilde{g}_{2,k}\\ \tilde{g}_{3,k}\end{bmatrix}. \tag{5.11}$$

The azimuth and elevation angles $\boldsymbol{\theta}_k = [\varphi_k, \vartheta_k]$ are given by trigonometry

$$\varphi[t] = \text{atan}\Big(\frac{y[t]}{x[t]}\Big), \tag{5.12}$$

$$\vartheta[t] = \arcsin\Big(\frac{z[t]}{\sqrt{x[t]^2+y[t]^2+z[t]^2}}\Big). \tag{5.13}$$

5.1 Reconstruction for NTF results

Now that there is a source reconstruction and a direction estimate available for every component, there are different ways of separating the sources. If the direction estimate is not used explicitly, the masks can be applied to the omnidirectional channel. In this case, the directional information was just used to inform the spectral separation. The advantage of this approach is that the sum of the sources resemble the omnidirectional channel when summed together and artefacts are easily masked when creating remixes

$$\underline{\mathbf{S}}_k = \frac{\mathbf{w}_k \mathbf{h}_k^T}{\mathbf{WH}} \circ \underline{\mathbf{X}}_1. \tag{5.14}$$

The other option is to use the directional estimate to direct a beamformer to the component directions. In this way, the estimated source also benefits from the spatial separation of the sources. The downside is that the sum of all components does not resemble the complete signal any more. So even with all extracted sources mixed together in mono, artefacts might be audible. The max-r_E weights described above can be applied here as well.

$$\underline{\mathbf{S}}_k = \frac{\mathbf{w}_k \mathbf{h}_k^T}{\mathbf{WH}} \circ (\underline{\mathbf{X}} \operatorname{diag}(\boldsymbol{a}) \boldsymbol{Y}(\boldsymbol{\theta}_k)) \tag{5.15}$$

Nevertheless, the spatial information can be used for clustering the components together, assuming that an instrument has no spatial extend. Note that the expression in brackets makes use of the inner product for tensors defined in 4.23.

Chapter 6

Component Clustering

After having completed a high dimensional separation with NMF or NTF, it is important to assign the K components to smaller number of groups, which represent the involved instruments. One option for component clustering, which is effective for both NMF and NTF, is based on the spectral properties of the dictionary entries, i.e. the columns of $\mathbf{W}$. In the NTF case, the spatial location can be utilized for clustering as well.

6.1 Clustering Algorithms

Clustering algorithms have the task of finding groups of similar data points in a d-dimensional feature space. Two different clustering algorithms are considered in this work.

Single Linkage. The single linkage algorithm is an hierarchical clustering scheme. It was used to create the dendrogram representations used for comparing MFCC with CQCCs, cf. Figure 6.3. It is a so called "agglomerative" clustering algorithm, where every data point starts out as it's own cluster. During convergence of the algorithm, these clusters are joined together to form a hierarchical structure where the distances between the clusters can be indicated. From these distances, a dendrogram representation can be created.

k-Means. The k-means algorithm is a partitioning clustering algorithm, which is based on dividing the feature space into cells, in which the data points are assigned to a cluster. The number of desired clusters has to be defined in advance, which is reasonable in the present application, since the number of desired instrument groups can be defined a-priori. In k-means, one random mean value for each cluster is chosen upon initialization. Then, the data points are assigned to the cluster with the closest mean, based on the Euclidean distance. After this, the means are re-computed for each cluster. These steps are repeated until convergence. k-means is well suited for clustering the spatial information, best done in the transformed, world-map representation, cf. Figure 7.8.

6.2 Cepstral Features

In speech processing, mel-frequency-cepstral coefficients (MFCCs) are commonly used features for discrimination tasks. Typically they are based on the STFT and computed as follows:

1. Compute STFT
2. Take the logarithm of the magnitude spectra
3. Apply mel-filters
4. Sum energies in each band over time
5. Apply the Discrete Cosine Transform (DCT)
6. Take the first (e.g. 13) coefficients

The logarithm accounts for the logarithmic human loudness perception and the mel-frequency filters for the pitch perception, which is assumed to be approximately linear at low frequencies and logarithmic at high frequencies. The discrete cosine transform (DCT) results in a decorrelation of the energy coefficients, after which a subset of the coefficients is enough for describing most of the information. It can be demonstrated that the DCT approximates the Karhunen-Loeve (KL) transform adequately, both for speech and music [Log00]. The DCT is a frequency transformation and the result to the inverse frequency transform of a logarithmic frequency representation is called cepstrum, hence the term cepstral coefficients.

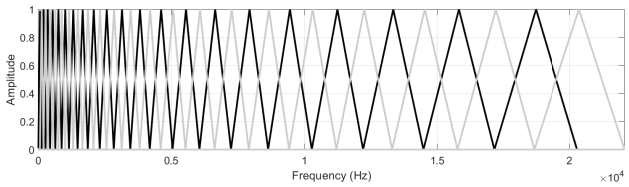


Figure 6.1 – Typical Mel Filterbank

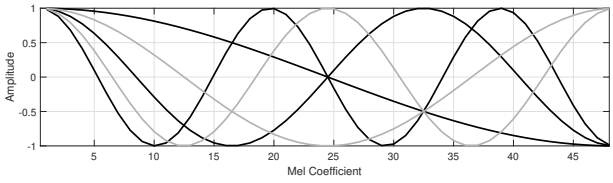


Figure 6.2 – First 5 DCT kernels

6.3 CQCCs

In the present case, when the CQT has been used as a transformation at the start, the existing mel-filterbank implementations can not be used. Although one can easily create a mel-filterbank, which takes the different frequency scale of the transform into account, the question arises, whether the logarithmic frequency resolution of the CQT itself might be suited to directly compute the cepstral coefficients without the effort of additional filtering. In particular, the used CQT implementation also features the regularisation parameter γ, which can be used to make the transform's frequency scale approximately linear at low frequencies, just as the mel scale does. Despite the criticism on the fit of the typically used mel-scale with experimental data [UCN88], MFCCs perform well in a technical environment. This might suggest that similar scales could also be suitable. In Figure 6.3, the Q-factor ($Q_f = \frac{\nu_f}{\Delta\nu_f}$) is compared for different scales.

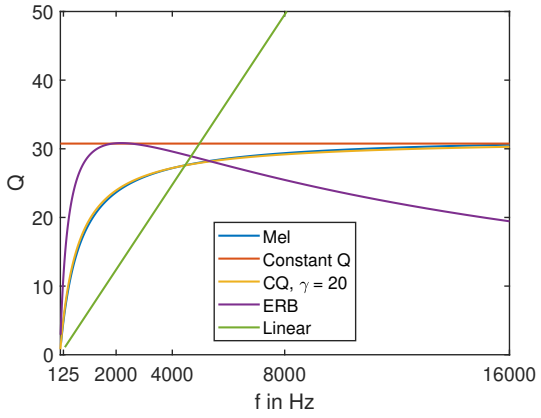


Figure 6.3 – Different frequency scales compared by $Q_f = \frac{\nu_f}{\Delta\nu_f}$. The CQT has a constant Q-factor by design. The CQT with regularisation $\gamma = 20$ is close to the mel scale. For best comparison, the frequency steps where chosen to approximately match the scale's maximal Q value (if it exists).

Discriminative Power. An application relevant criterion for choosing one of the scales, is how well the features that are based on it perform in a discrimination task. In figures 6.4 and 6.5, we see the result of cluster analysis of musical note events. The first example uses three piano chords and three bass notes. Both for CQCC and MFCC features, the two instrument groups can be distinguished, but the CQCC seems to perform better, as it exhibits larger distances between the clusters. The linear frequency scale fails to find two distinct clusters.

Also in a larger example with 4 instruments and 11 musical events, the CQCCs perform best. They manage to assign the notes to all 4 instruments correctly. From this small experiment, it can be concluded that CQCCs are a meaningful feature for finding instrument clusters.

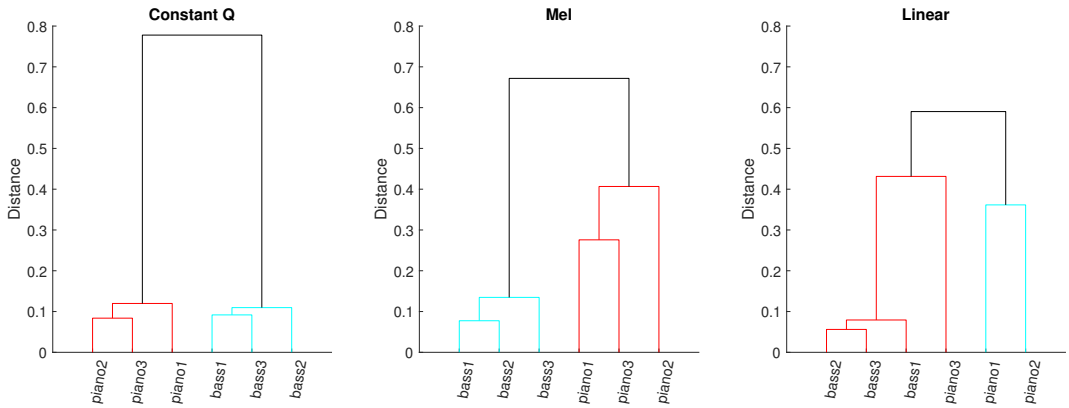


Figure 6.4 – Dendrogram representation of distances between found clusters for three bass notes and three piano chords, where cepstral features are based on different frequency scales. The constant Q scale seems to perform best in terms of separability.

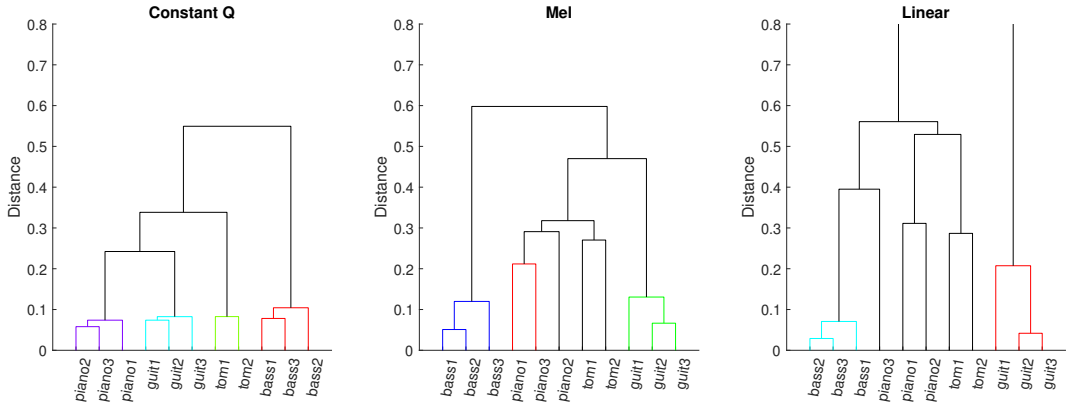


Figure 6.5 – Dendrogram representation of distances between found clusters for 4 different instruments comprised of 11 musical events in total. Again, the CQCCs perform best. Note that even though the distances between the clusters vary, the correct samples are next to each other in case of all three scales.

Chapter 7

Evaluation and Case Studies

7.1 Criteria: BSS Eval

To find useful criteria for the evaluation of source separation algorithms is a difficult task. The standard framework is based on the objective, energy-based criteria included in BSS Eval [VGF06]. These measures are for example used in the signal separation evaluation campaigns (SiSEC), which are carried out regularly. For the results from 2018, see [SLI18]. In the stereo separation tasks of these campaigns, the original criteria are extended by the Spatial Image Distortion Measure (SID), which describes how sources in a separated stereo signal are distorted spatially. In this work we are interested in extracting mono sources from stereo or Ambisonics mixes, so the original BSS Eval version is the best choice. The BSS Eval Matlab toolbox version 3 was used. The approach is based on splitting the estimated source signal $\hat{s}_k[t]$ into the following parts

$$\hat{s}_k[t] = s_{k,target}[t] + e_{k,interf}[t] + e_{k,noise}[t] + e_{k,artif}[t]. \tag{7.1}$$

Obviously, all true sources s_k and the noise component (if it exists) must be known in advance. The decomposition is done by projecting the signal vector onto different subspaces. The projection onto the subspace spanned by the true source is particularly straight forward, since the projection onto a single vector only involves the scalar product

$$s_{target}[t] = \frac{1}{||s_k[t]||^2}\Big[\sum_t \hat{s}_k[t]s_k[t]\Big]s_k[t]. \tag{7.2}$$

For projection to the interference subspace, more effort has to be taken, since the space spanned by all sources is not necessarily orthogonal. For projection onto a non-orthogonal space, the Gram-matrix $\boldsymbol{R}$ of the sources is required

$$r_{k,k'} = \sum_t s_k[t]s_{k'}[t]. \tag{7.3}$$

Let $\boldsymbol{s}[t] = [s_1[t], ..., s_K[t]]^T$ be a vector containing all true sources and $\boldsymbol{p}$ be a vector with the scalar products of the estimated source and all true sources

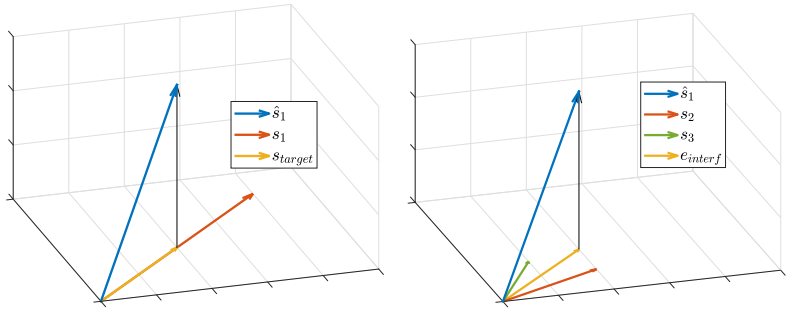
$$p_{k'} = \sum_t \hat{s}_k[t] s_{k'}[t]. \tag{7.4}$$

To obtain the interference signal the following projection is done

$$e_{interf}[t] = \boldsymbol{s}^T[t] \boldsymbol{R}^{-1} \boldsymbol{p} - s_{target}[t]. \tag{7.5}$$

The noise subspace only is relevant if a noise signal is given, which is typically not the case in the present separation scenarios, so that only the artifact signal constitutes the remaining part

$$e_{artif}[t] = \hat{s}[t] - s_{target}[t] - e_{interf}[t]. \tag{7.6}$$



(a) (b)

Figure 7.1 – Projection onto a vector, which resembles the corresponding true source signal and projection onto non-orthogonal subspace spanned by all true source signals.

Now that these signals are available, energy measures can be defined, which represent the separation's quality

$$\text{SDR} = 10\log \frac{\sum_t s^2_{target}[t]}{\sum_t e^2_{interf}[t] + e^2_{artif}[t]}, \tag{7.7}$$

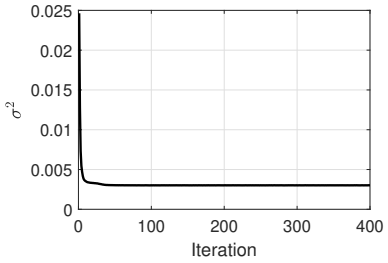
$$\text{SIR} = 10\log \frac{\sum_t s^2_{target}[t]}{\sum_t e^2_{interf}[t]}, \tag{7.8}$$

$$\text{SAR} = 10\log \frac{\sum_t s^2_{target}[t]}{\sum_t e^2_{artif}[t]}. \tag{7.9}$$

These criteria are a common first choice for comparing separation results, albeit it was noted that they can be modified to better correlate with the perceived quality [EVHH11].

7.2 Example 1: Bass/Guitar - Monaural

The files for the two examples presented in more detail here were taken from the "Mixing Secrets" free multitrack download library [cam], where perfectly separated instrument tracks could be obtained as ground truths. The first example shows the standard NMF Gibbs sampler on a simple bass/guitar separation. All prior parameters were set to zero in this first case. It can be seen that the algorithm converges after only a few iterations. Nevertheless, many iterations were performed for testing purposes.



Parameters	
K	7
F	361
N	4882
I	400
I_{burn}	200
α_0, β_0	0
λ_W, λ_H	0

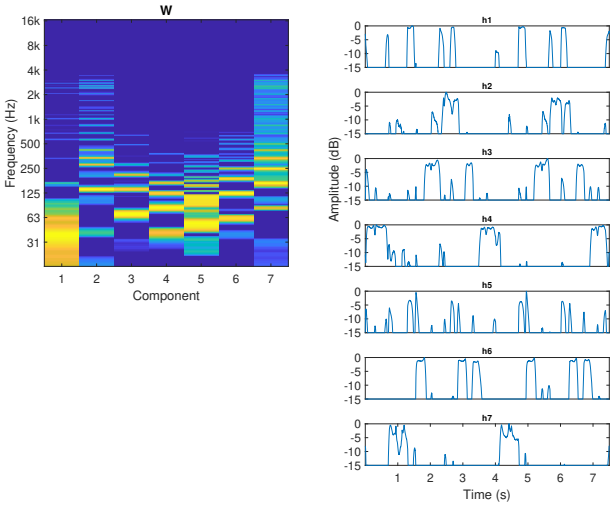


Figure 7.2 – Resulting dictionary matrix $\mathbf{W}$ and activation matrix $\mathbf{H}$. Both are normalized within the components and the $\mathbf{W}$ plot shows 30 dB dynamics.

For this task, no spatial information was available, so the clustering was solely based on the spectral information of the dictionary entries. The dendrogram representation of the proposed CQCC clustering is shown in Figure 7.3. Two clusters can easily be distinguished. The first one, featuring components 3, 6, 4 and 5, resembles the bass and the second one, comprised of components 2 and 7, corresponds to the guitar part. When looking at the dictionary itself, it becomes apparent that component 1 carries non-harmonic low-frequency content. It was assigned to the bass, which produced slightly better results than the assignment to the guitar or leaving out the component.

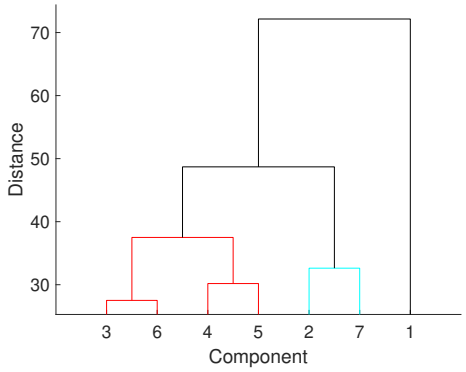


Figure 7.3 – Dendogram of the CQCC features from the dictionary matrix.

The quantitative separation results of this simple task are shown below. The interference measure (SID) is very high for both bass and guitar, which shows that the separation was successful. The separated guitar signal has a relatively low artefact measure (SAD), which also reduces the overall signal-to-distortion (SDR). When listening to the guitar separations, artefacts are audible, but made less obvious by the effects used on the guitar recording itself.

SDR		SID		SAR	
Bass	Guit.	Bass	Guit.	Bass	Guit.
17.3	2.6	25.7	14.0	18.0	3.1

Table 7.1 – BSS Eval Results for the first single channel separation task.

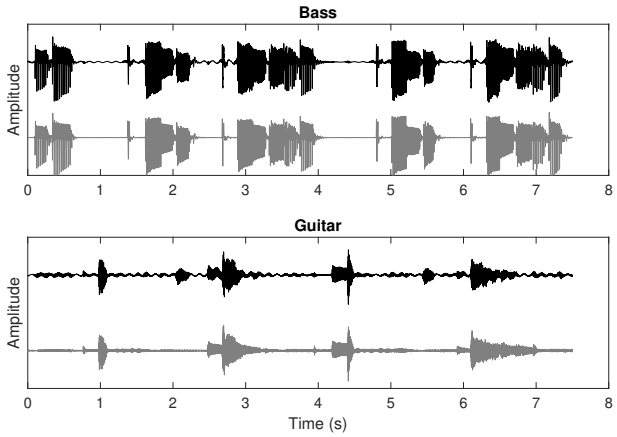


Figure 7.4 – Wavesforms of the separated sources (black) and the true sources (gray).

Handling Drums in the Multistage Approach. Now, a drum track was mixed in. To separate this, harmonic/percussive separation was used as a pre-processing step.

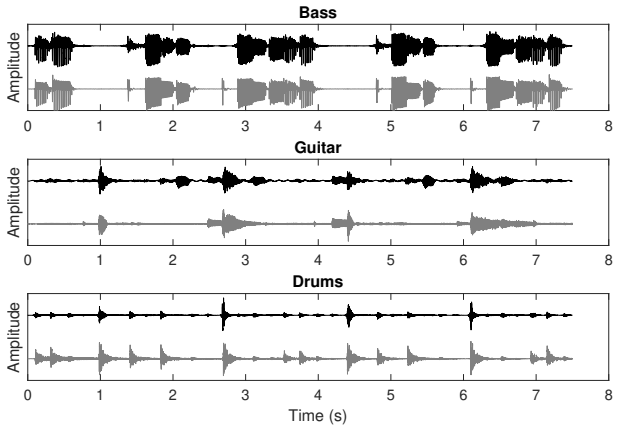


Figure 7.5 – Wavesforms of the separated sources (black) and the true sources (gray). Now, also a drum track was mixed in and extracted using the harmonic/percussive pre-processing described in chapter 3.

.

7.3 Example 2: Bass/Guitar/Piano - FOA

This example features an artificially created first order Ambisonics mix with 3 instruments: An electric guitar, a bass guitar and a piano. The guitar is panned to $\varphi = 90°, \vartheta = 0°$, the piano to $\varphi = -60°, \vartheta = 30°$ and the bass to $\varphi = 0°, \vartheta = -90°$. Bayesian NTF with Ambisonics processing has been applied to the first 3.6 seconds of the mix. 20 Iterations where performed using the following parameters.

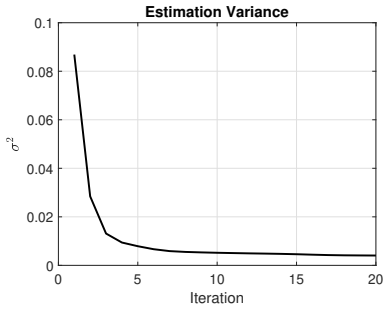


Figure 7.6 – Convergence behaviour.

Parameters	
K	10
F	361
N	2406
M	4
I	20
I_{burn}	10
λ_w	10
λ_h	1
α_0	100
β_0	0

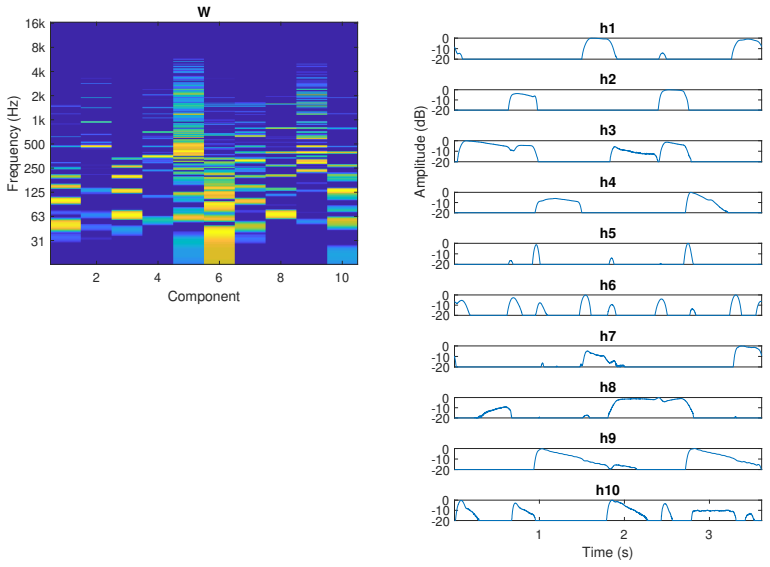


Figure 7.7 – Resulting dictionary matrix $\mathbf{W}$ and activation matrix $\mathbf{H}$, normalized in each column and row. 40 dB Dynamics plotted for $\mathbf{W}$.

This example demonstrates how the component direction estimation can make clustering very simple. Re-encoding the gain matrix and estimating the direction $\boldsymbol{\theta}_k$ yields the following component map.

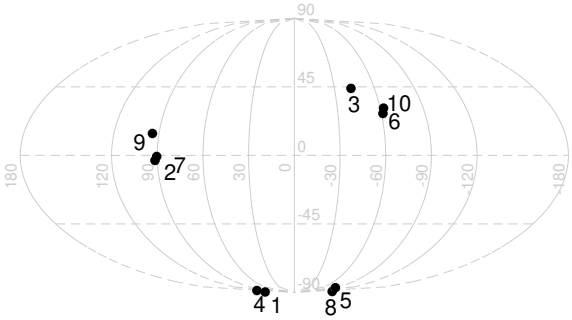


Figure 7.8 – Component directions determined from the gain matrix on a "world map" representation using mollweide-projection. The three apparent clusters correspond to the source directions, they can easily be found using k-means clustering.

BSS Eval Results. The table below shows the results of the quantitative evaluation, which reveals some interesting results. The first row represents the results obtained when clustering the components due to their spatial location and applying a the NTF mask to the output of a max-r_E beamformer directed towards the found directions. This complete process is called "NTF-Ambi" here. The second line shows the output of a max-r_E beamformer, assuming that the directions of the instruments are perfectly known, which is not the case in practical applications. The results show that for bass and guitar, the SDR for NTF-Ambi is improved over the beamforming solution by more then 10dB. Although the interference measure (SID) is strongly improved by applying the NTF result for all instruments, for the piano, the SDR is worse in case of applying both the beamformer and the mask. This is explained by a slight increase of artefacts, seen in the SAR value of NTF-Ambi. The last row shows the results of applying the beamformer to the found component directions, which consequently NTF could be used for as well (at a very high effort for the task). This shows that the found direction indeed correspond to the real ones. For the piano, the best result is obtained in this case, which might be caused by a beneficial position of the beamformer's zeros.

Distortion	SDR			SID			SAR		
Method	Bass	Piano	Guit.	Bass	Piano	Guit.	Bass	Piano	Guit.
NTF-Ambi	22.4	18.3	15.0	26.6	28.3	25.1	24.6	18.8	15.4
BF	10.8	22.7	5.7	10.8	22.8	5.7	42.8	38.4	38.0
BF-DE	10.9	28.5	6.6	10.9	28.9	6.6	43.0	38.7	38.4

Table 7.2 – BSS Eval measures for the full Ambisonics NTF approach, compared to using only a max-r_E beamformer in the exact source directions (assuming they are known) and a beamformer using the determined directions.

Chapter 8

Conclusion and Outlook

The fundamentals of NMF have been summarised and it's statistical viewpoint has been explained. It was shown that a Bayesian approach can yield successful source separation results in test examples, both with single and multichannel input. Particularly, separation based on a first-order Ambisonics input has been presented. The advantages of the statistical approach can be seen in the simple exchange of priors, for which the automatic relevance determination scheme serves as an example. Nevertheless, the separation results shown are still based on small toy examples and it is apparent that the task of improving spatial reproduction in practical situation is generally a difficult one.

For a practical algorithm, the separation would have to be done block-wise. In this case, the dictionary entries from the prior block could inform the prior densities of the next block. Future research might lead in the direction of doing Ambisonics source separation using combined NMF/CNN approaches or approaches fully based on Neural Networks.

Apart from the NMF aspects, the CQT has been revised and it could be shown to be effective in the source separation problem. The CQCC features, which are easily derived from it, are promising features for discrimination tasks, which might encourage using non-stationary spectrogram transformations even more. What has not been explicitly used here is the benefit that arises from the shift invariance of harmonics spectra when using the CQT, which is another clear advantage.

Also, the presented ambience extraction algorithm might be further explored for multichannel scenarios. Even a real-time implementation based on filter-banks could be interesting for the Ambisonics practise.

Nomenclature

$\underline{x}$	Complex numbers
$\underline{\mathbf{X}}$	Spectrogram matrix
$\mathbf{X}$	Magnitude spectrogram matrix
$\boldsymbol{\mathcal{X}}$	Magnitude spectrogram tensor
$\mathbf{W}$	First NMF factor matrix ("Dictionary")
$\mathbf{H}$	Second NMF factor matrix ("Activation")
$\mathbf{G}$	Gain matrix
$\mathbf{w}_k$	k-th column of the dictionary matrix
$\mathbf{h}_k^T$	k-th row of the activation matrix
$\theta, \boldsymbol{\theta}$	Estimated variable, matrix
$p(x\|y)$	Probability density function (pdf) of the random variable (RV) x, depending on the RV y
$p(x;y)$	pdf of the RV x with the parameter y
$x \sim p(y)$	RV x is distributed according to the density p
$\Phi(x)$	Cumulative density function (cdf)
$\mathcal{N}(\mu, \sigma^2)$	Normal pdf with parameters mean μ and variance σ^2
$\mathcal{TN}, \mathcal{E}, \mathcal{P}, \mathcal{G}$	Different pdfs, see below
$\boldsymbol{\theta}$	Direction vector, with azimuth φ and elevation ϑ
$\boldsymbol{y}(\boldsymbol{\theta})$	Spherical harmonics evaluated at $\boldsymbol{\theta}$, using ACN
$\boldsymbol{Y}(\boldsymbol{\Theta}_L)$	Matrix of spherical harmonics evaluated at $[\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_L]$

Appendix A

Definitions and Derivations

A.1 Involved pdfs

A.1.1 Gaussian

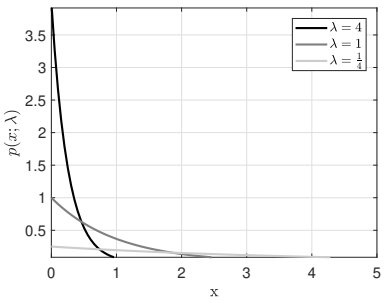
The multivariate Gaussian Distribution is defined as

$$\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^N det(\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}, \tag{A.1}$$

if the N random variables in the vector $\boldsymbol{x}$ are identically, independently distributed, all elements of the mean vector $\boldsymbol{\mu}$ are equal and the covariance matrix $\boldsymbol{\Sigma}$ becomes diagonal $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}$. Then every element is distributed according to

$$\mathcal{N}(x_n;\mu,\sigma^2) = \frac{1}{\sqrt{(2\pi\sigma^2)}} e^{-\frac{(x_n-\mu)^2}{2\sigma^2}}. \tag{A.2}$$

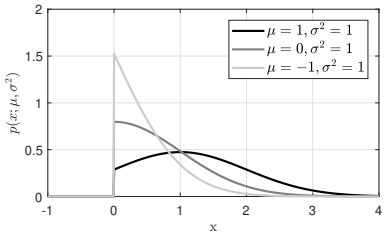
A.1.2 Exponential



$$\mathcal{E}(x;\lambda) = \lambda e^{-\lambda x} u(x) \tag{A.3}$$

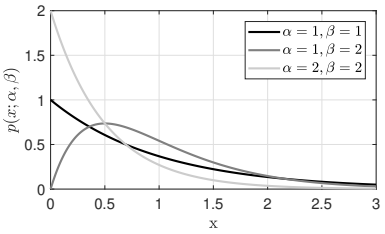
A.1.3 Truncated Gaussian

The particular model described in [SWH09] and in this thesis results in the product of a Gaussian and an exponential distribution as the factor matrix posteriors. As shown in equation A.26, this distribution can be described in terms of a truncated Gaussian, which is defined by



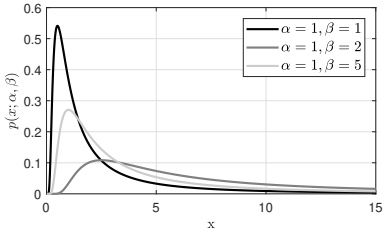
$$p(x; \mu, \sigma) = \mathcal{N}(x; \mu, \sigma)u(x) \tag{A.4}$$
$$= \mathcal{TN}(x; \mu, \sigma^2)_{[0,\infty)} \tag{A.5}$$

A.1.4 Gamma



$$p(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{(\alpha-1)} e^{-\beta x} \tag{A.6}$$

A.1.5 Inverse Gamma



$$p(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{(-\alpha-1)} e^{\frac{-\beta}{x}} \tag{A.7}$$

A.1.6 Poisson

The Poisson distribution is defined for integers only. It's probability mass function is equal to

$$P(x; \lambda) = e^{-\lambda} \frac{\lambda^x}{x!}. \tag{A.8}$$

A.2 Update Rule for the Euclidean Distance

One of the most compact derivation is found in [Bur14] and uses matrix calculus. We can formulate the squared Euclidean distance in terms of the Frobenius norm.

$$D(\mathbf{X}|\boldsymbol{W}\mathbf{H}) = \sum_{f=0}^{F-1}\sum_{n=0}^{N-1}(x_{fn} - y_{fn})^2 = ||\mathbf{X} - \mathbf{WH}||_F^2 \tag{A.9}$$

Now we can use the property which states that the trace of a product of matrices of the same dimension is equal to their element-wise product and write

$$\begin{aligned} D_{Euc} &= tr((\mathbf{X} - \mathbf{WH})^T(\mathbf{X} - \mathbf{WH})) && \text{(A.10)}\\ &= tr(\mathbf{X}^T\mathbf{X} - \mathbf{X}^T\mathbf{WH} - \mathbf{H}^T\mathbf{W}^T\mathbf{X} + \mathbf{H}^T\mathbf{W}^T\mathbf{WH}) && \text{(A.11)}\\ &= tr(\mathbf{X}^T\mathbf{X}) - tr(\mathbf{X}^T\mathbf{WH}) - tr(\mathbf{H}^T\mathbf{W}^T\mathbf{X}) + tr(\mathbf{H}^T\mathbf{W}^T\mathbf{WH}). && \text{(A.12)} \end{aligned}$$

Using gradient rules for the trace of a matrix, he quickly arrives at the gradient and consequently the update rules

$$\mathbf{W} \leftarrow \mathbf{W} - \boldsymbol{\eta}_{\mathbf{W}} \circ (\mathbf{WHH}^T - \mathbf{XH}^T) \tag{A.13}$$

$$\mathbf{H} \leftarrow \mathbf{H} - \boldsymbol{\eta}_{\mathbf{H}} \circ (\mathbf{W}^T\mathbf{HH} - \mathbf{W}^T\mathbf{X}). \tag{A.14}$$

The additive update equation turns into a multiplicative one when choosing

$$\boldsymbol{\eta}_{\mathbf{W}} = \frac{\mathbf{W}}{\mathbf{WHH}^T} \qquad\qquad \boldsymbol{\eta}_{\mathbf{H}} = \frac{\mathbf{H}}{\mathbf{W}^T\mathbf{WH}}, \tag{A.15}$$

because when expanding the expression, the left summand vanishes,

$$\mathbf{W} \leftarrow \mathbf{W} + \frac{\mathbf{W}}{\mathbf{WHH}^T} \circ (\mathbf{XH}^T - \mathbf{WHH}^T) = \mathbf{W} + \mathbf{W} \circ \frac{\mathbf{XH}^T}{\mathbf{WHH}^T} - \mathbf{W} \circ \frac{\mathbf{WHH}^T}{\mathbf{WHH}^T} \tag{A.16}$$

$$\mathbf{H} \leftarrow \mathbf{H} + \frac{\mathbf{H}}{\mathbf{W}^T\mathbf{WH}} \circ (\mathbf{W}^T\mathbf{X} - \mathbf{W}^T\mathbf{HH}^T) = \mathbf{H} + \mathbf{H} \circ \frac{\mathbf{W}^T\mathbf{X}}{\mathbf{W}^T\mathbf{WH}} - \mathbf{H} \circ \frac{\mathbf{W}^T\mathbf{WH}}{\mathbf{W}^T\mathbf{WH}}, \tag{A.17}$$

and the well known update eqations are obtained

$$\mathbf{W} \leftarrow \mathbf{W} \circ \frac{\mathbf{XH}^T}{\mathbf{WHH}^T} \tag{A.18}$$

$$\mathbf{H} \leftarrow \mathbf{H} \circ \frac{\mathbf{W}^T\mathbf{X}}{\mathbf{W}^T\mathbf{WH}}. \tag{A.19}$$

When taking a look at the gradients again, we see that in the final update equation, all negative terms $\nabla_{\boldsymbol{\theta}}^- D(\boldsymbol{\theta})$ are in the numerator and all the positive terms $\nabla_{\boldsymbol{\theta}}^+ D(\boldsymbol{\theta})$ are in the denominator. This rule holds for the derivation of other multiplicative update equations as well

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} \circ \frac{\nabla_{\boldsymbol{\theta}}^- D(\boldsymbol{\theta})}{\nabla_{\boldsymbol{\theta}}^+ D(\boldsymbol{\theta})} \tag{A.20}$$

A.3 Derivation of the Gibbs Sampler

A.3.1 NMF case

Factor Posteriors. The most important step for the derivation of the Gibbs sampler is to separate the specific w_{fk}, from which we wish to sample, from the sum over all components. Also, all terms that do not include w_{fk} are proportional constants and can be neglected. Normalization can be carried out separately for a correct analytical expression of the pdf, which is not important for sampling from it

$$p(w_{fk}|\mathbf{X}, \mathbf{W}_{\setminus(w_{fk})}, \mathbf{H}, \sigma^2) = p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \sigma^2)p(w_{fk}; \lambda^W_{fk}) \tag{A.21}$$

$$\propto \prod_n \mathcal{N}(\mathbf{X}|\mathbf{WH}, \sigma^2)\mathcal{E}(w_{fk}; \lambda^W_{fk})u(w_{fk}) \tag{A.22}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_n (x_{fn} - \sum_k w_{fk}h_{kn})^2\Big)\exp(-\lambda^W_{fk}w_{fk})u(w_{fk}) \tag{A.23}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_n (x_{fn} - w_{fk}h_{kn} - \sum_{k'\neq k} w_{fk'}h_{k'n})^2\Big)\exp(-\lambda^W_{fk}w_{fk})u(w_{fk}) \tag{A.24}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_n (-2x_{fn}w_{fk}h_{kn} + w^2_{fk}h^2_{kn} + 2w_{fk}h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda^W_{fk}w_{fk}\Big)u(w_{fk}) \tag{A.25}$$

$$\propto \exp\Big(-w^2_{fk}\frac{1}{2\sigma^2}\sum_n h^2_{kn} + w_{fk}\big(\frac{1}{\sigma^2}\sum_n (x_{fn}h_{kn} - h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda^W_{fk}\big)\Big)u(w_{fk}). \tag{A.26}$$

The resulting distribution can be expressed in terms of a Gaussian with mean μ_W and variance σ^2_W, truncated on $[0, \infty)$

$$\mathcal{N}(w_{fk}|\mu_W, \sigma^2_W) \propto \exp\Big(-\frac{1}{2\sigma^2_W}(w_{fk} - \mu_W)^2\Big)u(w_{fk}) \tag{A.27}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2_W}w^2_{fk} + \frac{1}{\sigma^2_W}w_{fk}\mu_W\Big)u(w_{fk}). \tag{A.28}$$

Comparing coefficients of the exponent-polynomial in w_{fk} between eq. A.26 and eq. A.28, mean μ_W and variance σ^2_W are found to be

$$\sigma^2_W = \frac{\sigma^2}{\sum_n h^2_{kn}} \tag{A.29}$$

$$\mu_W = \sigma^2_W\big(\frac{1}{\sigma^2}\sum_n (x_{fn}h_{kn} - h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda^W_{fk}\big). \tag{A.30}$$

The derivation of the sampler for h_{kn} is very similar

$$p(h_{kn}|\mathbf{X}, \mathbf{W}, \mathbf{H}_{\setminus h_{kn}}, \sigma^2) = p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \sigma^2)p(h_{kn}; \lambda_{kn}^H) \tag{A.31}$$

$$\propto \prod_f \mathcal{N}(\mathbf{X}|\mathbf{WH}, \sigma^2)\mathcal{E}(h_{fkn}; \lambda_{kn}^H)u(h_{kn}) \tag{A.32}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f (x_{fn} - \sum_k w_{fk}h_{kn})^2\Big)\exp(-\lambda_{kn}^H h_{kn})u(h_{kn}) \tag{A.33}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f (x_{fn} - w_{fk}h_{kn} - \sum_{k'\neq k} w_{fk'}h_{k'n})^2\Big)\exp(-\lambda_{kn}^H h_{kn})u(h_{kn}) \tag{A.34}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f (-2x_{fn}w_{fk}h_{kn} + w_{fk}^2h_{kn}^2 + 2w_{fk}h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda_{kn}^H h_{kn}\Big)u(h_{kn}) \tag{A.35}$$

$$\propto \exp\Big(-h_{kn}^2\frac{1}{2\sigma^2}\sum_f w_{fk}^2 + h_{kn}\big(\frac{1}{\sigma^2}\sum_f (x_{fn}h_{kn} - w_{fk}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda_{kn}^H\big)\Big)u(h_{kn}) \tag{A.36}$$

$$\propto \exp\Big(-\frac{1}{2\sigma_H^2}(h_{kn} - \mu_H)^2\Big)u(h_{kn}). \tag{A.37}$$

The posterior of h_{kn} can be described in terms of a Gaussian, truncated on the interval $[0, \infty)$ with mean and variance equal to

$$\sigma_H^2 = \frac{\sigma^2}{\sum_f w_{fk}^2} \tag{A.38}$$

$$\mu_H = \sigma_H^2\big(\frac{1}{\sigma^2}\sum_f (x_{fn}w_{fk} - w_{fk}\sum_{k'\neq k} w_{fk'}h_{k'n}) - \lambda_{kn}^H\big). \tag{A.39}$$

Variance Posterior. The inverse gamma distribution is the conjugate prior for a Gaussian likelihood with unknown variance. This means that the posterior belongs to the same family of distributions, i.e. the posterior is distributed according to the inverse gamma distribution as well. The posterior for σ^2 is given by

$$p(\sigma^2|\mathbf{X}, \mathbf{W}, \mathbf{H}) = p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \sigma^2)p(\sigma^2) \tag{A.40}$$

$$\propto \prod_f \prod_n \mathcal{N}(\mathbf{X}|\mathbf{WH}, \sigma^2)\mathcal{G}^{-1}(\sigma^2; \alpha, \beta) \tag{A.41}$$

$$\propto \frac{1}{(2\pi\sigma^2)^{\frac{NF}{2}}}\exp\Big(\sum_f\sum_n -\frac{1}{2\sigma^2}(x_{fn} - \sum_k w_{fk}h_{kn})^2\Big)(\sigma^2)^{-\alpha-1}\exp\Big(-\frac{\beta}{\sigma^2}\Big) \tag{A.42}$$

$$\propto (\sigma^2)^{-\alpha-1-\frac{NF}{2}}\exp\Big(-\frac{1}{\sigma^2}\big(\frac{1}{2}\sum_f\sum_n (x_{fn} - \sum_k w_{fk}h_{kn})^2 - \beta\big)\Big) \tag{A.43}$$

$$\propto (\sigma^2)^{-\alpha'-1}\exp\Big(-\frac{\beta'}{\sigma^2}\Big), \tag{A.44}$$

with the parameters

$$\alpha' = \alpha + \frac{NF}{2} \tag{A.45}$$

$$\beta' = \frac{1}{2}\sum_f\sum_n (x_{fn} - \sum_k w_{fk}h_{kn})^2 + \beta. \tag{A.46}$$

A.3.2 NTF case

Derivation for $\mathbf{G}$ with exponential prior analogous to the NMF factors.

$$p(g_{km}|\mathbf{X},\mathbf{W},\mathbf{H},\mathbf{G}_{\setminus g_{km}}),\sigma^2) = p(\mathbf{X}|\mathbf{W},\mathbf{H},\mathbf{G},\sigma^2)p(g_{km};\lambda^G_{km}) \tag{A.47}$$

$$\propto \prod_f \prod_n \mathcal{N}(x_{fnm}|\sum_k w_{fk}h_{kn}g_{km},\sigma^2)\mathcal{E}(g_{km};\lambda^G_{km})u(g_{km}) \tag{A.48}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(x_{fnm}-\sum_k w_{fk}h_{kn}g_{km})^2\Big)\exp(-\lambda^G_{km}g_{km})u(g_{km}) \tag{A.49}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(x_{fnm}-w_{fk}h_{kn}g_{km}-\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})^2\Big)\exp(-\lambda^G_{km}g_{km})u(g_{km}) \tag{A.50}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(-2x_{fn}w_{fk}h_{kn}g_{km}+w^2_{fk}h^2_{kn}g^2_{km} \tag{A.51}$$

$$+2w_{fk}h_{kn}g_{km}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})-\lambda^G_{km}g_{km}\Big)u(g_{km}) \tag{A.52}$$

$$\propto \exp\Big(-g^2_{km}\frac{1}{2\sigma^2}\sum_f\sum_n w^2_{fk}h^2_{kn} \tag{A.53}$$

$$+g_{km}\big(\frac{1}{\sigma^2}\sum_f\sum_n(x_{fn}w_{wf}h_{kn}-w_{fk}h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})-\lambda^G_{km}\big)\Big)u(g_{km}) \tag{A.54}$$

Comparing coefficients to a truncated Gaussian again, the mean and the variance take the form

$$\sigma^2_G = \frac{\sigma^2}{\sum_f\sum_n w^2_{fk}h^2_{kn}} \tag{A.55}$$

$$\mu_G = \sigma^2_G\big(\frac{1}{\sigma^2}\sum_f\sum_n(x_{fnm}w_{fk}h_{kn}-w_{fk}h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})-\lambda^G_{km}\big). \tag{A.56}$$

The derivation of the $\mathbf{W}$ and $\mathbf{H}$ update rules are again very similar, with the sums gaining one new dimension. The resulting means and variances of the truncated Gaussian posteriors are

$$\sigma^2_W = \frac{\sigma^2}{\sum_m\sum_n h^2_{kn}g^2_{km}} \tag{A.57}$$

$$\mu_W = \sigma^2_W\big(\frac{1}{\sigma^2}\sum_m\sum_n(x_{fnm}h_{kn}g_{km}-h_{kn}g_{km}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})-\lambda^W_{fk}\big), \tag{A.58}$$

$$\sigma^2_H = \frac{\sigma^2}{\sum_m\sum_f w^2_{fk}g^2_{km}} \tag{A.59}$$

$$\mu_H = \sigma^2_H\big(\frac{1}{\sigma^2}\sum_m\sum_f(x_{fnm}w_{fk}g_{km}-w_{fk}g_{km}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})-\lambda^H_{kn}\big). \tag{A.60}$$

The derivation for $\mathbf{G}$ with uniform prior is equivalent to setting $\lambda_G = 0$ and imposing an additional limit at $g_{km} = 1$

$$p(g_{km}|\mathbf{X}, \mathbf{W}, \mathbf{H}, \mathbf{G}_{\setminus g_{km}}), \sigma^2) = p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \mathbf{G}, \sigma^2)p(g_{km}) \tag{A.61}$$

$$\propto \prod_f \prod_n \mathcal{N}(x_{fnm}|\sum_k w_{fk}h_{kn}g_{km}, \sigma^2)\mathcal{U}(g_{km;0,1}) \tag{A.62}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(x_{fnm} - \sum_k w_{fk}h_{kn}g_{km})^2\Big)u(g_{km})(1-u(g_{km}-1)) \tag{A.63}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(x_{fnm} - w_{fk}h_{kn}g_{km} - \sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})^2\Big)u(g_{km})(1-u(g_{km}-1)) \tag{A.64}$$

$$\propto \exp\Big(-\frac{1}{2\sigma^2}\sum_f\sum_n(-2x_{fn}w_{fk}h_{kn}g_{km} + w_{fk}^2h_{kn}^2g_{km}^2 \tag{A.65}$$

$$+ 2w_{fk}h_{kn}g_{km}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})\Big)u(g_{km})(1-u(g_{km}-1)) \tag{A.66}$$

$$\propto \exp\Big(-g_{km}^2\frac{1}{2\sigma^2}\sum_f\sum_n w_{fk}^2h_{kn}^2 \tag{A.67}$$

$$+ g_{km}\big(\frac{1}{\sigma^2}\sum_f\sum_n(x_{fn}w_{wf}h_{kn} - w_{fk}h_{kn}\sum_{k'\neq k} w_{fk'}h_{k'n}g_{k'm})\big)\Big)u(g_{km})(1-u(g_{km}-1)). \tag{A.68}$$

Bibliography

[AJ02] C. Avendano and J. M. Jot, "Ambience extraction and synthesis from stereo signals for multi-channel audio up-mix," in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, May 2002, pp. II–1957–II–1960.

[AJ04] C. Avendano and J.-M. Jot, "A frequency-domain approach to multichannel upmix," *Journal of the Audio Engineering Society. Audio Engineering Society*, vol. 52, pp. 740–749, 07 2004.

[BDJ+11] P. Balazs, M. Dörfler, F. Jaillet, N. Holighaus, and G. Velasco, "Theory, implementation and applications of nonstationary gabor frames," *j. Comput. Appl. Math.*, vol. 236, no. 6, pp. 1481–1496, Oct 2011.

[BFL17] T. Brouwer, J. Frellsen, and P. Lio, "Comparative study of inference methods for bayesian nonnegative matrix factorisation," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Jan 2017, pp. 513–529.

[BL17] Z. Botev and P. L'Ecuyer, "Simulation from the normal distribution truncated to an interval in the tail," in *Proceedings of the 10th EAI International Conference on Performance Evaluation Methodologies and Tools*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2017, pp. 23–29.

[Bro91] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[Bur14] J. J. Burred, "Detailed derivation of multiplicative update rules for nmf," Tech. Rep., March 2014. [Online]. Available: https://www.semanticscholar.org/paper/Detailed-derivation-of-multiplicative-update-rules-Burred/3376b4df752f2428c451e530f9c6e0ce3a3f05e4

[cam] "The 'mixing secrets' free multitrack download library." [Online]. Available: http://www.cambridge-mt.com/ms-mtk.htm

[CDA18] E. K. Canfield-Dafilou and J. S. Abel, "A group delay-based method for signal decorrelation," in *Audio Engineering Society Convention 144*, May 2018.

[Cem09] A. Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Computational Intelligence and Neuroscience*, vol. 2009, 2009.

[Cla17] J. Clarke, "Source separation in action: Demixing the beatles at the hollywood bowl," in *Audio Engineering Society Convention 142*, May 2017.

[Coy09] E. Coyle, "On the use of the beta divergence for musical source separation," *IET Conference Proceedings*, pp. 34–34(1), Jan 2009.

[Dan00] J. Daniel, "Représentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimédia," Ph.D. dissertation, Université Paris, Jan 2000. [Online]. Available: http://www.theses.fr/2000PA066581

[Doe01] M. Doerfler, "Time-frequency analysis for music signals: A mathematical approach," *Journal of New Music Research*, vol. 30, pp. 3–12, 03 2001.

[ES08] B. Eisenberg and R. Sullivan, "Why is the sum of independent normal random variables normal?" *Mathematics Magazine*, vol. 81, no. 5, pp. 362–366, 2008.

[EVHH11] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, "Subjective and objective quality assessment of audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2046–2057, Sept 2011.

[FBD09] C. Fevotte, N. Bertin, and J. Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, March 2009.

[FCC05] D. Fitzgerald, M. Cranitch, and E. Coyle, "Non-negative tensor factorisation for sound source separation," in *Proceedings of Irish Signals and systems conference*, 2005.

[Fit10] D. Fitzgerald, "Harmonic/percussive separation using median filtering," *13th International Conference on Digital Audio Effects (DAFx-10)*, Jan 2010.

[Fit11] ——, "Upmixing from mono - a source separation approach," in *2011 17th International Conference on Digital Signal Processing (DSP)*, July 2011, pp. 1–7.

[FJCR11] D. Fitzgerald, R. Jaiswal, E. Coyle, and S. Rickard, "Shifted nmf using an efficient constant-q transform for monaural sound source separation," in *22nd IET Irish Signals and Systems Conference 2011*, June 2011.

[Gab47] D. Gabor, "Theory of communication," *Journal of the Institution of Electrical Engineers - Part I: General*, vol. 94, no. 73, pp. 58–, January 1947.

[GL84] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, April 1984.

[HDB11] R. Hennequin, B. David, and R. Badeau, "Beta-divergence as a subclass of bregman divergence," *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 83–86, Feb 2011.

[HDVG13] N. Holighaus, M. Dörfler, G. A. Velasco, and T. Grill, "A framework for invertible, real-time constant-q transforms," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 775–785, April 2013.

[Hoy02] P. O. Hoyer, "Non-negative sparse coding," in *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, vol. cs.NE/0202009, 2002.

[LL09] R. Liu and S. Li, "A review on music source separation," in *2009 IEEE Youth Conference on Information, Computing and Telecommunication*, Sept 2009, pp. 343–346.

[Log00] B. Logan, "Mel frequency cepstral coefficients for music modeling," *Proc. 1st Int. Symposium Music Information Retrieval*, Nov 2000.

[LS99] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[LS01] ——, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. MIT Press, 2001, pp. 556–562.

[MBD16] P. Magron, R. Badeau, and B. David, "Phase recovery in NMF for audio source separation: an insightful benchmark," in *40th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. abs/1605.07469, 2016.

[Mol08] C. B. Moler, "Numerical Computing with Matlab," in *Numerical Computing with MATLAB, Revised Reprint*. Philadelphia: SIAM, 2008, ch. 9.3.

[NZDS11] C. Nachar, F. Zotter, E. Deleflie, and A. Sontacchi, "Ambix - a suggested ambisonics format," in *3rd International Symposium on Ambisonics and Spherical Acoustics*, Jun 2011.

[Rob95] C. P. Robert, "Simulation of truncated normal variables," *Statistics and Computing*, vol. 5, no. 2, pp. 121–125, Jun 1995.

[Roh15] L. Rohr, "Evaluation of audio source separation in the context of 3d audio," Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, 2015.

[Rou15] L. Roux, "Sparse nmf - half-baked or well done?" Mitsubishi Electric Research Laboratories, Cambridge, Tech. Rep., 2015.

[SD06] S. Sra and I. S. Dhillon, "Generalized nonnegative matrix approximations with bregman divergences," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. C. Platt, Eds. MIT Press, 2006, pp. 283–290.

[SK10] C. Schörkhuber and A. Klapuri, "Constant-q transform toolbox for music processing," in *Proceedings of the 7th Sound and Music Computing Conference, Barcelona, Spain*, July 2010.

[SLI18] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *Latent Variable Analysis and Signal Separation: 14th International Conference, LVA/ICA 2018, Surrey, UK*, 2018, pp. 293–305.

[SWH09] M. N. Schmidt, O. Winther, and L. K. Hansen, "Bayesian non-negative matrix factorization," in *Independent Component Analysis and Signal Separation*, T. Adali, C. Jutten, J. M. T. Romano, and A. K. Barros, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 540–547.

[TF13] V. Y. F. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization with the /spl beta/-divergence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1592–1605, July 2013.

[UCN88] S. Umesh, L. Cohen, and D. Nelson, "Fitting the mel scale," in *IEEE conference on Acoustics, Speech, and Signal Processing*, vol. 1, 04 1988, pp. 217 – 220 vol.1.

[VCG08] T. Virtanen, A. T. Cemgil, and S. Godsill, "Bayesian extensions to nonnegative matrix factorisation for audio signal modelling," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2008, pp. 1825–1828.

[VGF06] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, July 2006.

[VHDG11] G. A. Velasco, N. Holighaus, M. Doerfler, and T. Grill, "Constructing an invertible constant-q transform with nonstationary gabor frames," in *Proceedings of the 14th International Conference on Digital Audio Effects, DAFx 2011*, Sept 2011.

[VTG08] T. Virtanen, A. Taylan Cemgil, and S. Godsill, "Bayesian extensions to nonnegative matrix factorisation for audio signal modelling," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2008, pp. 1825–1828.

[ZF12] F. Zotter and M. Frank, "All-round ambisonic panning and decoding," *J. Audio Eng. Soc*, vol. 60, no. 10, pp. 807–820, 2012.

[ZF19] ——, *Ambisonics - A Practical 3D Audio Theory for Recording, Studio Production, Sound Reinforcement, and Virtual Reality*, first edition ed. Heidelberg (Deutschland): Springer, May 2019.