University of West Bohemia
Faculty of Applied Sciences

Doctoral Thesis

2013 Ing. Štěpán Albrecht

University of West Bohemia

Faculty of Applied Sciences

Doctoral Thesis

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in specialization

Computer Science and Engineering

Štěpán Albrecht

Model-based Approaches for Automatic Transcription of Music

Supervisor: Prof. Ing. Václav Matoušek, CSc.

Department of Computer Science and Engineering

Pilsen, 2013

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Disertační práce

k získání akademického titulu doktor

v oboru Inženýrská informatika

Štěpán Albrecht

Modelově-orientované přístupy pro automatickou hudební transkripci

Školitel: Prof. Ing. Václav Matoušek, CSc.

Katedra: Informatiky a výpočetní techniky

Plzeň 2013

Prohlášení

Předkádám tímto k posouzení a obhajobě disertační práci zpracovanou na závěr doktorského studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji tímto, že tuto práci jsem vypracoval samostatně, s použitím odborné literatury a dostupných pramenů uvedených v seznamu, jenž je součástí této práce.

V Plzni dne 28. srpna 2013 Štěpán Albrecht

Acknowledgments

This work was made at Faculty of Applied Science at University of West Bohemia in Pilsen between years 2005 – 2013. For distributed computations, I highly acknowledge the access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the program "Projects of Large Infrastructure for Research, Development, and Innovations" (LM2010005).

I would like to thank to my supervisor Prof. Václav Matoušek for providing me with the opportunity to research in the music signal processing topic, and his guidance; then I would like to express great thanks to Václav Šmídl for his expert advice in mathematical models. I also appreciate useful suggestions of Tomáš Pavelka and Jakub Červený during my PhD studies and to the thesis itself.

Lastly I cannot forget to express great thanks to my parents for their patience with me, especially during the time when I was completing the thesis, and also great thanks to my friends for their support in situations, that sometimes might have been difficult.

Abstract

The problem of memory based complete automatic music transcription is considered. The complete automatic music transcription, i.e., estimation of (i) all sounds in time, (ii) their instrumentation and (iii) their loudnesses, is a difficult and in some cases even not solvable problem. Even though the three named music content features carry the entire information for the original music signal composition, they can represent observed data for further processing, e.g., of the music piece tempo as another music content feature. Therefore the practical complete automatic music transcription follows a scenario – an intention – and tries to capture all the features within the scenario. In this work, the inverse music sequencer as a specific scenario for the complete automatic music transcription is defined. A monoaural music signal and the library of sounds as an input of the inverse music sequencer is considered. The sounds in the library are to be composed of harmonic sounds (a piano, a flute, ...) and drum sounds. A probabilistic model containing unobserved variables which reflect information of truncation parameters of library sounds sought in the observed signal their displacements in time and their amplitudes is designed. The detection of subparts of the library sounds is a distinct feature of our approach in comparison to other approaches that consider only full sequences of frames. Variational Bayes method to calculate equations of estimates of the unobserved variables is applied. Evaluation methods for the specific intention of the inverse music sequencer are introduced. In the experimental part, the sensitivity analysis respecting an observed music signal, library of sounds, nuisance parameters and various modifications of the transcription algorithm is carried out. In experiments, one sound library contains harmonic sounds of one music instrument, thus music instrument recognition is not a part of our experiments although the proposed transcription algorithms are developed for this too.

Abstrakt

Disertační práce se zabývá problémem úplné automatické hudební transkripce. Úplná automatická hudební transkripce, tj. detekce (i) všech zvuků v čase, (ii) nástrojů jejich reprodukce a (iii) jejich hlasitostí, je složitý a v některých případech dokonce teoreticky neřešitelný problém. I když zmíněné tři charakteristiky hudebního obsahu nesou úplnou informaci k reprodukování skladby, někdy tvoří jen data pro další zpracování, např. pro získání tempa skladby jako další charakteristiky hudebního obsahu. Proto se úplná hudební transkripce omezuje na scénář – záměr – v rámci kterého usiluje o zachycení všech charakteristik. V této práci definujeme inverzní hudební sekvencer jako tento scénář. Mono-audio hudební signál a knihovna (banka zvuků) tvoří vstupní data inverzního hudebního sekvenceru. V knihovně mohou být nahrávky harmonických zvuků (piano, flétna, ...), zvuky bicích nástrojů, případně celé nahrávky jimi tvořené. Navrhujeme pravděpodobnostní model, jehož odhadované proměnné nesou informaci o parametrech zkrácení knihovních zvuků hledaných ve vstupním hudebním signálu, jejich rozmístění v čase a jejich amplitudách. Detekce podčástí knihovních zvuků je vlastnost, kterou detekujeme jen ve scénáři našeho inverzního hudebního sekvenceru, jiné postupy pracují se zvukem jako s celkem. Pro výpočet neznámých proměnných je aplikována variační Bayesovská technika. Zavádíme metody vyhodnocování pro scénář inverzního hudebního sekvenceru. V části "Experimenty" provádíme citlivostní analýzu v závislosti na vstupním hudebním signálu, knihovně zvuků, volných parametrech modelu a různých modifikacích transkripčního algoritmu. Jedna knihovna zvuků v našich experimentech obsahuje pouze zvuky – tóny jednoho harmonického hudebního nástroje, a tak rozpoznávání hudebních nástrojů není součástí testů, i když navržené transkripční algoritmy jsou vhodné i pro něj.

List of Abbreviations

AMT	Automatic music transcription
DFT	Discrete Fourier transform
STFT	Short-time Fourier transform
F0	Fundamental frequency
ASA	Audio scene analysis
MFCC	Mel-frequency cepstral coefficients
ICA	Independent component analysis
PCA	Principal component analysis
NMF	Non-negative matrix factorization
ISA	Independent subspace analysis
MIDI	Music instrument digital interface
EM	Expectation-maximization
GMM	Gaussian mixture model
HMM	Hidden Markov model
MIR	Music information retrieval
SDR	Sound-to-distortion ratio
SNR	Sound-to-noise ratio
SIR	Sound-to-interference ratio
SAR	Sound-to-artifact ratio
ML	Maximum likelihood
MAP	Maximum a posteriori
MMSE	Minimum mean square error
MCMC	Monte Carlo Markov chain
KL	Kullback-Leibler
VB	Variational Bayes
IVB	Iterative variational Bayes
EKF	Extended Kalman filter
o-bank	Library of sounds, the observed signal was created from this.
e-bank	Library of sounds used in estimation process.
unsp.	Unspecified (Table 4.1)
NM	Not meaningful (Table 4.1)
SL	Sound Library (Section 4.5)
SD	Simulation Data (Section 4.5)
VST	Virtual studio technology (technology for software synthesizers by Steinberg corporation)

Contents

Chapter 1

Introduction

Automatic music transcription (AMT) is a process of analysis of an acoustic music signal so as to write down the pitch, onset time, duration and source of each sound that occurs in it [1]. In Western tradition, written music uses *note symbols* to indicate these parameters in a piece of music. The note symbols are contained in a *music score*. Another parameter resulting from the AMT can be the loudness[1] of a sound source. Besides the common musical notation, the transcribed products can take many other forms. E.g., *chord symbols* are usually sufficient for a guitar player to describe his role in an orchestra. DJs operate with *tempo* ("speed" of a music piece) and *meter key* (number of note lengths on a *bar*). A *genre classification* can be utilized for indexing of music. In a transcription system, a MIDI[2] file is often an appropriate format for musical notations.

A complete AMT – i.e., resolving pitch, loudness, timing and instrumentation of all sound events in an input audio music signal – is very difficult or even not theoretically possible in some cases [1], therefore the goal of practical AMT is redefined as being able to notate as many of constituent sounds as possible (complete AMT) or to transcribe some well-defined part of the music signal, for example, the dominant melody or the most prominent drum sound (partial AMT). The complete AMT follows a specific scenario.

As a scenario, the auditory scene analysis (ASA) in music signals can be considered (Kashino et al. [4]). The ASA aims at extracting entities like notes and chords from an audio signal. Sound source models (low level) operate on algorithms devised from psychophysical findings regarding the acoustic "clues" that humans use to assign the spectral components to their respec-

[1] Contrary to note symbols the volume of loudness is specified for larger parts in a music score.

[2] Musical Instrument Digital Interface (MIDI) is a standard format for exchanging performance data and parameters between electronic musical devices [2, 3].

tive sources. Musicological[3] models (higher level) are also applied. Another scenario represents the music scene description (Goto et al.) [5], where the aim is to obtain descriptions that are intuitively meaningful to an untrained listener without trying to extract every musical note from input music signal. This includes the analysis of melody, bass lines, metrical structure, rhythm and chorus and phrase repetition. The last example of scenarios concerns the signing transcription. The system of Ryynänen et al. [6] is capable to convert a recorded singing into a sequence of discrete notes and their starting and ending points in time. It consists of two stages – low level – estimation of continuous pitch track, higher level – segmentation of the pitch track into discrete note events and quantizing their pitch values. They utilized the framework of the Gaussian mixture model with the hidden Markov model (GMM / HMM) [7].

Another division of the AMT considers two classes: memory-based and data-based AMT. The former utilizes sound models corresponding to certain musical instrument sounds, therefore it can be used to identify instruments. The latter utilizes only rules which hold in general, e.g., harmonic sounds have most prominent magnitude spectrum peaks approximately in k-multiplies of their *fundamental frequency* (F0).

1.1 Terminology

The following terms are defined in [1]. *Timbre* is a term for "sound color". *Pitch* represents the perceived *fundamental frequency* of a sound. While actual fundamental frequency (F0) can be precisely determined through physical measurement, it may differ from perceived pitch because of *overtones* (*or partials*) in the sound. The partials are frequencies of higher intensity which change the sound timbre. *Tone* is a representation of a sound having detectable pitch. Tones are written as notes in a *score*. Given the reference fundamental frequency of a tone, one octave frequency range is a multiple of two of the reference fundamental frequency. There are 12 tones in one *octave*. In normal tuning, the closest upper tone frequency to a tone of frequency $\text{frequency}_{\text{ref}}$ is given by $2^{1/12} \cdot \text{frequency}_{\text{ref}}$. A harmonic tone is a tone having its partials (called *harmonics* here) approximately in the k-multiplies of its fundamental frequency. Harmonic sounds are produced, e.g., by a piano, violin, acoustic guitar. *Musical key* of a piece usually refers to the first note of a chord, which gives a subjective sense of arrival and rest of a music piece. *Musical meter* refers to rhythmic patterns produced by grouping together

[3]Musicological models include relations between music content events, e.g., note transitions, chord tones in given a music key, etc.

strong and weak beats. The meter may be duple (2 beats in a measure), triple (3 beats in a measure), quadruple (4 beats in a measure) and so on.

1.2 Signal Representations

Discrete Fourier transform (DFT), Mel-frequency cepstral coefficients (MFCC), chroma, frames, spectrogram (overview in [1]): the DFT produces a frequency representation of a stationary sound. A modification of the DFT is short-time Fourier transform (STFT) [8]. Given any music signal, the STFT takes that signal, segment by segment, applies windowing function[4] and calculates the DFT. This way we obtain a time-frequency representation of an audio / music signal – a spectrogram, its vectors over time are called frames. The MFCCs reflect the timbral aspects of the music signal. They are calculated on a segment, like the DFT, so that the segment sound is processed by a discrete cosine transform (the imaginary part of the DFT is disregarded), the obtained coefficients are summed on the frequency logarithmic Mel-scale, the resulting sums are forwarded to logarithm and processed by an inverse DFT. Chroma is an approximated 12-dimensional vector of simultaneously-sounding pitches irrespective of octaves. It is used as a feature vector for music signal pitch similarity definition.

1.3 Approaches to One-channel Automatic Music Transcription

Most audio recordings can be viewed as mixtures of several audio signals, called source signals, which are usually active simultaneously. The sources may have been mixed synthetically with a mixing console or by recording a real audio scene using microphones. If the number of microphones is greater or equal to the number of sources, then the unsupervised separation of convolutive mixtures in time domain or mixtures without convolution in a complex frequency domain enables, theoretically, a perfect separation of sources [9]. The imperfections can be caused by recording devices or audio data quantization. If the number of microphones is smaller, the problem is *underdetermined.* We can still utilize the unsupervised convolutive separation methods [9], but various techniques like sparse coding or statistical approaches need to be utilized as aids [9, 10, 11]. Usually, a separation of a real audio signal is a strongly underdetermined problem – the observational data are picked

[4] The windowing function reduces side-lobes in the resulting spectrum – frame.

up from one or two microphones. Information of the AMT can be obtained by additional processing of the separated sources (see Unsupervised Learning Methods).

In the following paragraphs, approaches to one-channel automatic music transcription are presented. They can be divided into two groups: those using an observation signal model of source superposition (unsupervised learning methods, statistical methods (1), (2) and those which do not (the remaining). Author's proposed solution operates on the former set. Much of the designer's efforts with the signal model methods is spent on finding such representations of equations whose space of unobserved variables is not large.

Computational Models of Human Auditory System

At the present time the ears and the brain of a trained musician are the most reliable music transcription system available. Compared with any artificial audio processing tool, the analytical ability of human hearing is very good for complex mixture signals: in natural acoustic environments, we are able to perceive the characteristics of several simultaneously occurring sounds including their pitches [12]. These computational models follow the operations of (i) confirmed physical processes of human ear and (ii) presumed processing of human brain. The former are represented by filtering using filters spread equally on a logarithmic scale, then the model of inner-hair cells is followed. The inner hair cells model is described by a sequence of the following operations: compression, half-wave rectification and lowpass filtering of the filter outputs [13]. The latter is represented by autocorrelation function [14] or a combination of adaptive oscillators and neural networks [15]. The resulting algorithms are focused on F0-calculations [14, 15, 16] or acoustic feature extraction (MFCC coefficients), e.g., for onset detection [17] or music sound classification ([1], Chapter 6; [18]). The outer and inner ear processes are utilized to enhance the sound separation problems solution, e.g., by an emphasis on low frequencies of the input signal [19].

Auditory Scene Analysis

ASA refers to the human capability to perceive and recognize individual sound sources in mixture signals [1]. It can be viewed as a two stage process: first, the audio signal is transformed into a time-frequency representation, second, the components (bins) of the time-frequency plain are grouped into their respective sound sources. In humans, the grouping stage has been found to depend on various acoustic properties ("clues") of the components, such as their harmonic frequency relationships, common onset times, or synchronous

frequency modulation [12]. The work of Brown and Godsmark [20] and the work of Kashino et al. [4, 21] can be named as examples. Kashino's transcription system uses a memory bank of harmonic sounds to identify musical instruments and to help the ASA with grouping techniques. It also uses chord-note relations, chord transition relations and perceptual rules. Its ASA part and the musicological models are connected in a Bayesian network. Currently, this work represents one of the most elaborate music transcription systems.

Unsupervised Learning Methods

These methods do not utilize the source-specific prior knowledge and learn sources from a long segment of an audio signal. They are determined by the following (observed) signal model representing a superposition of sources:

$$\boldsymbol{y}_\tau \approx \boldsymbol{F}\boldsymbol{g}_\tau, \tag{1.1}$$

Here $\boldsymbol{y}_\tau$, $\tau = 1, \ldots, t$ denotes the observed audio signal – input of the method. Output of the unsupervised learning methods is the memory sound source bank $\boldsymbol{F}$ of size $\phi \times k$ (denoted here as the *library of sounds*) whose columns represent sound sources. The second output is the vector of gains $\boldsymbol{g}_\tau$ of the sources. The quantities $\boldsymbol{y}_\tau$ or $\boldsymbol{F}$ are represented either by the magnitude or by the power spectrum (spectra). It is possible to represent $\boldsymbol{y}_\tau$ by time-domain data. In this case, the matrix $\boldsymbol{F}$ contains sines and cosines of the due frequency values [22], similar signal models for unsupervised learning methods in the time domain can be seen in [1, 23]. The complete AMT can be carried out by postprocessing of the unsupervised learning method product $\boldsymbol{F}$, $\boldsymbol{g}_\tau$: the instruments are identified by classification techniques between the detected source library $\boldsymbol{F}$ and the memory bank labeled by instrument names (see [1], Chapter 6), if the detected sources represent monophonic recordings, the pitch detection becomes a simple problem. The gain $\boldsymbol{g}_\tau$ determines a presence of a source for the unsupervised learning methods to be applied. It must be held: $k \ll t$ and in order to provide a reasonable sound separation, k must be given beforehand.

One of the unsupervised learning methods is the independent component analysis (ICA). It assumes that the elements of the vector $\boldsymbol{y}_\tau$ are independent and non-Gaussian. The core of the ICA algorithm carries out the estimation of an unmixing matrix $\boldsymbol{W} \approx \boldsymbol{F}^{-1}$ to result in the source vector estimate $\hat{\boldsymbol{g}}_\tau = \boldsymbol{W}\boldsymbol{y}_\tau$ where the estimated sources are denoted by $\hat{\boldsymbol{g}}_\tau$. The matrix $\boldsymbol{W}$ is estimated so that the rows of the output data matrix $\hat{\boldsymbol{G}} = [\hat{\boldsymbol{g}}_1, \ldots, \hat{\boldsymbol{g}}_t]$ are

maximally independent. In order to ensure an approximate independence of the ICA input data rows $[\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t]$, the *whitening* needs to be performed:

$$\tilde{\boldsymbol{y}}_\tau = \boldsymbol{U}(\boldsymbol{y}_t - \boldsymbol{\mu}), \tag{1.2}$$

where $\tilde{\boldsymbol{y}}_\tau$ is the whitened vector, $\boldsymbol{U}$ is the whitening matrix obtained from the eigenvalue decomposition and $\boldsymbol{\mu}$ is the empirical mean.

Remark. The whitening [24, 25] is the decorrelation method. When $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t$ are regarded as realizations of a random quantity of a covariance matrix $\boldsymbol{\Sigma}$, the whitening transforms them into a set of new random quantity whose covariance is $a.\boldsymbol{I}$, where a is a scalar value and $\boldsymbol{I}$ is the identity matrix. The new random variables are uncorrelated and have their variance equal to 1. The whitening (1.2) by the eigenvalue decomposition is termed as the principal component analysis (PCA) [1, 26]. Besides the decorrelation, the PCA is capable to reduce the dimensionality by omitting small eigenvalues and the due eigenvectors. Another whitening procedure called "scaling" may be seen in Chapter 3.

After the ICA and calculation of $\hat{\tilde{\boldsymbol{g}}}_\tau$ with the decorrelated $\tilde{\boldsymbol{y}}_\tau$, the inverse of the decorrelation on $\hat{\tilde{\boldsymbol{g}}}_\tau$ must be applied. The ICA algorithm input and output data can be both the time and frequency domain data. The standard ICA [27] is not aimed at underdetermined problems – in one-channel tasks the vector $\tilde{\boldsymbol{y}}_\tau$ would become a scalar value instead. Consider the data $\boldsymbol{Y}^T = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t]^T$, $\boldsymbol{G}^T$ are represented by a magnitude or the power spectrum. Then the ICA with whitening is suitable for the separation of sources. Such approach is referred to as the independent subspace analysis (ISA) [1, 28, 29]. The musical components, we look for, are not mutually independent, therefore, after we obtain $\hat{\boldsymbol{G}}^T$ from the ICA algorithm, its rows need to be grouped into the musical components. The grouping requires measuring of dependencies between the rows. This is termed as the multidimensional ICA [30].

In the case of magnitude and power spectra in $\boldsymbol{Y}$, it is advantageous to restrict the representation of sources to be entry-wise non-negative and also not to allow negative gains. Even though the non-negative ICA has been studied, e.g., in [31], the standard ICA does not allow the non-negativity to impose on $\boldsymbol{Y}, \boldsymbol{G}, \boldsymbol{F}$. Lee and Seung proposed two cost functions and estimation algorithms [32, 33, 34] to obtain $\boldsymbol{Y} \approx \boldsymbol{FG}$. The algorithm is called non-negative matrix factorization (NMF). Here the meaning of $\boldsymbol{F}$ and $\boldsymbol{G}$ is swapped – the cost functions for gains $\boldsymbol{G}$ and sources $\boldsymbol{F}$ are given by Euclidean distance:

$$d_{euc}(\boldsymbol{Y}, \boldsymbol{FG}) = \sum_{n,\tau} ([\boldsymbol{Y}]_{n,\tau} - [\boldsymbol{FG}]_{n,\tau})^2 \tag{1.3}$$

and the divergence

$$d_{div}(\boldsymbol{Y}, \boldsymbol{FG}) = \sum_{n,\tau} D([\boldsymbol{Y}]_{n,\tau}, [\boldsymbol{FG}]_{n,\tau}), \tag{1.4}$$

where the divergence[5] $D(p, q)$ is defined as

$$D(p, q) = p \log \frac{p}{q} - p + q. \tag{1.5}$$

The NMF estimation algorithms iteratively minimize:

$$\boldsymbol{F} \leftarrow \boldsymbol{F}. * (\boldsymbol{X}\boldsymbol{G}^T)./(\boldsymbol{F}\boldsymbol{G}\boldsymbol{G}^T), \tag{1.6}$$

$$\boldsymbol{G} \leftarrow \boldsymbol{G}. * (\boldsymbol{F}^T\boldsymbol{X})./(\boldsymbol{F}^T\boldsymbol{F}\boldsymbol{G}) \tag{1.7}$$

for the Euclidean distance and

$$\boldsymbol{F} \leftarrow \boldsymbol{F}. * (\boldsymbol{X}./\boldsymbol{F}\boldsymbol{G})\boldsymbol{G}^T./(\boldsymbol{1}\boldsymbol{G}^T), \tag{1.8}$$

$$\boldsymbol{G} \leftarrow \boldsymbol{G}. * \boldsymbol{F}^T(\boldsymbol{X}./\boldsymbol{F}\boldsymbol{G})./(\boldsymbol{F}^T\boldsymbol{1}) \tag{1.9}$$

for the divergence. Here $\boldsymbol{1}$ is $\phi \times t$ matrix having all its elements equal to one and $.*$, $./$ denote element-wise multiplication and division, respectively. In the NMF algorithm initialization step, the matrices $\boldsymbol{F}$, $\boldsymbol{G}$ are initialized by random positive values. As noted in [1], Chapter 9, factorization of the magnitude spectrogram using the divergence often produces relatively good results, and according to [35] the NMF with this divergence outperforms the NMF with the Euclidean distance. The divergence cost of an individual observation $[\boldsymbol{Y}]_{n,\tau}$ is linear as a function of the scale of the input, since $D(\alpha p, \alpha q) = \alpha D(p, q)$ for any positive scalar α whereas for the Euclidean cost the dependence is quadratic. The NMF algorithm for the divergence has been used in multi-pitch detection [36, 37], drum transcription [38] and sound separation tasks [18, 19, 35, 39, 40, 37]. When the drum transcription is considered, usually one column vector $\boldsymbol{f}_s$ of the source matrix $\boldsymbol{F}$ suffices to represent a drum sound. When the multi-pitch transcription is considered, one vector $\boldsymbol{f}_s$ represents one pitch, e.g., in [37] of Smaragdis et al. Since the spectra of the sources can vary a lot, Vincent et al. in [41] improved the NMF of Smaragdis for multi-pitch detection by a special representation of source vectors: he represented a source vector as a linear combination of narrowband harmonic and inharmonic spectra having the bands logarithmically placed in

[5]The divergence reminds in some aspects the Kullaback-Leibler (KL) divergence (see in Chapter 2), but the meaning of the KL divergence is different – it operates with distributions not with data elements.

frequency domain. Virtanen proposed a convolutive model where one source is modeled as a spectrogram $\tilde{\boldsymbol{F}}_s \in \boldsymbol{\mathcal{F}}$. The columns of the spectrogram represent a sequence of stretchings of one source vector of a pitched musical instrument [23]. In [36], he improved this model by a modification that each core source vector $\tilde{\boldsymbol{f}}_s$ is represented by a linear combination of "excitations" (i.e., by another source basis vectors), hence, the number of source vectors in the model was reduced again.

Statistical Methods

If a source-specific prior knowledge is to serve as an aid in a parameter identification, it is convenient and often necessary to use the Bayesian framework, see in Chapter 2. The statistical methods occur in all aforementioned approaches. They can be divided into:

1. Methods using the signal model (1.1) in its observation distribution $p(\boldsymbol{y}_\tau|\boldsymbol{F}, \boldsymbol{g}_\tau, \boldsymbol{\psi})$, where $\boldsymbol{\psi}$ denotes hyperparameters of the probabilistic model. Probability models allow to incorporate various knowledge of the music signal behavior, e.g., a sparsity on gains which yields the log-posterior density

$$\log p(\boldsymbol{G}|\boldsymbol{Y}, \boldsymbol{F}) = -\lambda \sum_{n,\tau} ([\boldsymbol{Y}]_{n,\tau} - [\boldsymbol{F}\boldsymbol{G}]_{n,\tau})^2 - \sum_{n,\tau} \log p([\boldsymbol{G}]_{n,\tau}) \quad (1.10)$$

   and a temporal continuity (i.e., if a sound is present at time τ, it is possibly present at time $\tau + 1$ too) [35][6]. The gains $\boldsymbol{G}$ can be obtained from (1.10) by gradient descent methods [42], the sources $\boldsymbol{F}$ can be learned from $p(\boldsymbol{Y}', \boldsymbol{F}) = \int_{\boldsymbol{G}^*} p(\boldsymbol{Y}'|\boldsymbol{G}, \boldsymbol{F}) p(\boldsymbol{G}) d\boldsymbol{G}$ by gradient descend methods, too. Here $\boldsymbol{Y}'$ can represent either the observed data (unsupervised approach) $\boldsymbol{Y}$ or the training musical audio data (supervised approach). This approach was presented in [43], for instance. Another approach [44] uses a signal model, where, similarly to [36], a source tone spectrum is represented by a linear combination of basis spectra. The basis spectra are learned from training data. In the set of supervised AMT approaches there are models whose hyperparameters of prior sources $\boldsymbol{\psi}$ are learned instead of spectra [22, 45, 46]. Should be noted that the AMT algorithms mostly operate on harmonic sounds in their observed data.

[6] We refer to [35], where its signal model is not expressed in terms of probability model, however it allows reformulation in terms of probability model [39].

2. Methods of Smaragdis et al. which use multinomial distributions (see Appendix) in source separation algorithms [47, 48]. The magnitude spectrograms of observed and training data are concerned to be drawn from these distributions. Their signal model of superposition differs of the signal model (1.1) – the number of draws from the distribution of combined signal is equal to a sum of draws from distributions of sources, thus it contains no gain term $\boldsymbol{g}$. The probabilistic model of the whole separation is given by:

$$P_t(f) = \sum_s P_t(s) \sum_{z \in \{\boldsymbol{z}_s\}} P_s(f|z) P_t(z|s). \tag{1.11}$$

where $P_t(f)$ is the probability of observing frequency f in time frame t in the mixture spectrogram; $P_s(f|z)$ is the probability of frequency f in the z-th learned basis vector from source s; $P_t(z|s)$ is the probability of observing the z-th basis vector of source s at time t; $\{\boldsymbol{z}_s\}$ represents the set of values the latent variable z can take for source s; and $P_t(s)$ is the probability of observing source s at time t. The distribution terms corresponding to each training set are learned by an EM algorithm [49]. The reconstruction itself is performed by another EM algorithm. If observed data is long enough, the algorithm does not need the learned reconstruction term $P_s(f|z)$ to get a reasonable result – the separation can be performed unsupervised. The sparse coding in the form of an entropic prior is applied to decrease a number of "active" elements in speaker-dependent mixture weight distributions $P_t(z|s)$ and the source priors $P_t(s)$. Even though the methods have been published mainly to solve the source separation problem, they could be utilized for pitch and instrument identification [50].

3. Methods using the GMM / HMM framework adopted, e.g., from the automatic speech recognition. It occurs, e.g., in the singing transcription system [6], here each considered MIDI tone was represented by a three-state HMM model. The features were collected from the difference between the estimated fundamental frequency candidates[7] and the model tone fundamental frequency, along with the onset attack parameters. The transitions of the HMMs were trained with Baum-Welch algorithm [7] on a set of annotated singing data.

4. Pattern recognition approaches for music instrument recognition [52]

[7] They were calculated by computational model of human auditory system of Klapuri [51].

or music genre classification [53] may be statistical, too[8] – linear discriminative analysis, Bayes classifier, GMM.

5. Bayesian network in the AMT system of Kashino et al., see the ASA approach above.

1.4 Motivation

Our scenario is to design an algorithm that can identify arbitrary sounds in observed music signals. In order for this task to be accomplished, the prior knowledge for sources – memory bank – must be passed to the algorithm. Since they are arbitrary, some subpart of a sound of the memory bank can be perceptually similar to another sound from the memory bank. E.g., one sound can be a tone C_1 of a piano and another sound can be a sequence of piano tones C_1 – G_1. Moreover, there are more types of pianos, they can be recorded under various acoustic conditions. Our aim is to cover various instrumental properties in order to reduce the number of sounds in the memory bank but still to allow music content identification. E.g., if our algorithm manages to identify a subpart of the memory bank sound with the observed audio, we do not have to keep both C_1 and C_1 – G_1 in the bank. If the algorithm manages a pitch shift of a sound, then in the memory bank there can be just a few pitches of tones of an instrument. The memory bank and the labeling of its sounds specify the purpose of the algorithm, i.e., of the transcription. If its purpose is, for instance, the multiple fundamental frequency estimation, a library sound representing a tone can be combined from more instruments and loudnesses so that the observed signal timbre does not affect the estimation. Whereas if the purpose is the flute and piano identification, the sound library should not contain other instrument sounds.

Intuitively, our formalization of the problem can be understood as an "inverse music sequencer", Fig. 3.1. Music sequencers have a pre-recorded library of sounds (sound components) which are combined together to create a music signal. The input to the sequencer is a MIDI file which contains information about the beginning of music events in time, their duration, IDs of sounds (in our case the pre-recorded sound components), their amplitude and modification type. In the proposed solution of this thesis, we consider only component truncation as a possible modification[9]. The output of the

[8] The music instrument recognition or genre classification is not only a matter of statistical approaches, but also a matter of approaches as (i) k-nearest neighbors, neural networks (see an overview in [1], Chapter 6) or (ii) support vector machines [54].

[9] Another modifications not considered in this thesis are, e.g., strech/shrink of a library sound resulting in its pitch shift.

sequencer is an audio signal. The input of our "inverse music sequencer" is a recorded music signal and its output is the estimated (transcribed) MIDI-like representation of music events.

The truncation also imposes some restrictions on sounds which are identified and identification methods themselves. The sounds which evolve in time, are also allowed to be a part of the sound library. E.g., a drum loop of a typical drum set or a sequence of notes of a harmonic instrument represent feasible sounds of the library. However, gaining a general time-varying sound for the bank to allow the identification in the observed signal can be difficult. Consider, e.g., a sound of thunder or flowing water or a DJ's scratch of a playing record. It can be complicated to record them "twice the same". The structural differences between the sounds of the observed recording and the library sounds can be overcome by a lesser number of sounds in the library.

There have been approaches designed so as to identify arbitrary sounds (not just harmonic or drum) which may overlap [55] using the GMM / HMM approach, however, they do not allow the identification of the truncations of the library sounds. We used statistical approaches allowing us to specify a signal model of the superposition of more segments (frames) from sounds in the library. The additional restrictions of the statistical model allow the detection of the truncation parameters and reduce the number of free parameters in the model. In Subsection 1.3, we recall that harmonic musical instruments and drums are suitable to be identified by the signal model approaches. In our experiments, however, we deal only with harmonic sounds, since the number of results is already large.

1.5 Applications

Current AMT applications encompass: music recommendation which, based on set of songs, can recommend other songs; "Hit Song Science": a tool which claims to reliably measure the hit potential of novel songs; querying a search engine for music by, e.g., a hummed melody instead of typing a text; audio fingerprinting: based on an excerpt of a music song this music song can be identified in a large database of songs; various music information retrievals: genre, notation, control of lights in discotheques; plug-ins for the alignment of singing imperfections in pitch. A survey of the applications can be found in [1, 56].

We will discuss applications following the proposed motivation. With a reasonable library of sounds being labeled, the resulting algorithm can detect information of instrumentation, pitch and displacement of all music events. Obtained information can be either the objective of our transcription or

can be used further on, e.g., as feature vector(s) for identification purposes. Since it provides a complete AMT, its utilization can be large. We divide applications of our AMT approach in the following three types:

Audio Coding

The purpose of audio coding is to reduce the size of musical data while the quality of musical recording is retained to some degree. The output of Section 1.4 – Motivation – is a type of a MIDI representation. MIDI file is an extremely compact representation of musical data by its music content. The coding of MIDI-like information is called the *structured audio coding* [57] and it is implemented in MPEG-4 standard [58]. In the MPEG-4 standard, only the MIDI-like information is coded. The decoder uses a standardized library of sounds to combine it with the decoded MIDI-like information.

Analysis and Manipulation

Algorithm based on our motivation performs an analysis. The analysis can provide multi-pitch detection or instrumentation recognition. A user, having analyzed a recording of a drum loop or an instrument melody in such a way, can change the instrumentation, arrangement or loudness of a particular instrument and re-synthesize the recording thereafter.

Preprocessing Step for Subsequent Music Information Retrieval

Algorithm based on our motivation can provide features for subsequent analysis resulting in music information retrieval (MIR). Most of the current MIR applications named in the beginning of this section utilize, or could utilize MIDI-like features instead of features obtained directly from the audio signal.

1.6 State-of-the-art in Automatic Music Transcription

A reliable complete AMT system does not presently exist. Up to this date, transcription capabilities of skilled human musicians outperform any music transcription system in accuracy and flexibility. However, some degree of success has been achieved for polyphonic music of limited complexity. In the transcription of pitched and percussive (drum) instruments, typical restrictions are that the number of consecutive sounds is limited, interference

of drums and percussive sounds is not allowed in the recording, or only a specific instrument is considered.

Some promising results for the multi-pitch transcription of real-world CD music recordings (containing drums and non-harmonic sounds too) have been demonstrated by Goto [59], Ryynänen and Klapuri [60]. In the former example, Goto et al. detected pitch of a melody and pitch of a bass line. The tested music recordings were hand-labeled by using their own developed tool; they refer 88% and 80% of an average detection rate for the melody and bass line, respectively. In the latter case, Ryynänen extended his singing transcription system [6] using the multi-pitch detector of Klapuri [51] on arbitrary polyphonic real music recordings; unlike Goto et al., he did not utilize tone model spectra, but trained the transition probabilities of the HMM model. He defined the evaluation rules as follows: A reference note is correctly transcribed by a note in the transcription if (i) their MIDI notes are equal, and (ii) the absolute difference between their onset times is smaller than or equal to a given maximum onset interval given beforehand, and (iii) the transcribed note is not already associated with another reference note. He refers to recall[10] of 39%, precision of 41%, and mean overlap ratio[11] 40%. Unfortunately, it is not much clear how the testing data were labeled.

In multipitch transcription the results are better if only harmonic instrument(s) are present in the observed music signal. We recall the performance of transcription approaches described in Subsection 1.3. All of them were evaluated on simulated observed data. Abdallah in [43] evaluated his approach by matching note-on events for each pitch to those in the original MIDI file, to within a tolerance of about 46 ms which was the STFT frame length on 11025 Hz, too. Of his evaluation set, 94.3% notes were correctly detected, while 2.2% of the triggered (note) onsets were "false-positives" that did not match any note in the original. Marolt's transcription system [15] achieves, for synthesized piano music, average detection and false-positive rates of 90% and 9%, respectively, improving to 98% and 7% for a synthesized Bach partita. Marolt used a set of adaptive oscillators to obtain the features reflecting the fundamental frequency thus did not work with a fixed spectrum frame length. In the transcription of tone superposition of various harmonic instruments, Klapuri [16] quotes an "error rate" of 9% for two note chords, rising to 28% for four note chords in case of 46 ms long frames, and 6% for two note chords, rising to 12% for four note chords in case of 96 ms STFT on 44.1 kHz. The multipitch detector of Vincent et al. based on the

[10]The evaluation concept of precision, recall and F-measure is widely used in evaluation of AMT problems. It is described in our proposed solution evaluation in Section 4.6.

[11]The overlap ratio refers to a measure of a detected length of a note to the ground truth length.

NMF algorithm [41] yielded a value of 87% in the F-measure of correct detected notes. Kashino's transcription system [4] resulted in 92.5% for flute – piano note recognition and 77.3% for flute – piano – clarinet. The evaluation percentage of [4] was provided in R-index:

$$R = \left(\frac{right - wrong}{total} \cdot \frac{1}{2} + \frac{1}{2} \right) \cdot 100\%. \tag{1.12}$$

The observation signal characteristics and its time-frequency analysis parameters were not denoted there.

In drum transcription without interference of harmonic sounds, Virtanen and Paulus achieved 95% in recognition of bass-drum, snare-drum and hihats on real recorded patterns [38] using the NMF approach. In this NMF approach, a particular drum sound was represented by one source vector. The source vector was calculated by a linear combination of particular drum sound sources from training data. Performance of these approaches is summarized in Table 4.1.

The signal models are utilized either for the pitched or drum transcription; or, for the sound source separation methods. The evaluation of sound source separation is a type of evaluation for music transcription tasks. The concept is referred to as sound-to-distortion ratio (SDR) measure (4.3) and it is characterized in Chapter 4 – Experiments. In [39], Virtanen et al. compare an approach of Bayesian extension of the NMF to the previous NMF approach in [35]. In [35] for pitched instruments: they denote 25% source detection error and $SDR = 9.8dB$, and for drums: 22% source detection error and $SDR = 6.0dB$. In [39] for pitched instruments: they denote 28% source detection error and $SDR = 12.3dB$, and for drums: 20% source detection error and $SDR = 6.0dB$. The testing mixtures were prepared the same way for both approaches: by allotting a number of sources randomly (pitched up to 12, drums up to 6), allotting a random length of pitched sounds, random amplitude, a random number of drum sound repetitions and onset time. Note that the evaluation concept of SDR was extended by Gribonval et al. by introducing sound-to-artifacts (SAR), sound-to-interference (SIR) and sound-to-noise ratio (SNR) [61]. The SDR represents a total measure comprising SAR, SIR and SNR.

1.7 Thesis Objectives and Outline

The thesis objectives are as follows:

- Definition of the inverse music sequencer as a scenario for the complete automatic music transcription. A monoaural observed music signal

considered for the complete automatic music transcription can be composed of harmonic and drum sounds.

- Design of a probabilistic model with unobserved variables which reflect information of truncation parameters of library sounds presented in the observed signal, their displacements in time and their amplitudes. Application of variational Bayes method to calculate formulas of estimates of the unobserved variables.

- Introduction of evaluation methods for the scenario of the inverse music sequencer.

- Experimental part:

  - Sensitivity analysis with respect to an observed music signal, library of sounds, nuisance parameters and various modifications of the transcription algorithm. In experiments, one sound library contains harmonic sounds of one musical instrument, thus musical instrument recognition is not a part of our experiments although the proposed transcription algorithms are developed for this too.
  - The evidence that the proposed solution can outperform or compete with the state-of-the-art in multi-pitch detection which represents a setup option of the inverse music sequencer.

The thesis outline is as follows: Chapter 1 contains an overview of approaches to automatic music transcription with attention to transcription of pitched and drum sounds solved by the estimation on the basis of signal models and statistical approaches. The state-of-the-art of the approaches containing the values of accuracy are presented here along with the inverse music sequencer scenario in Section 1.4. Chapter 2 introduces the necessary probability theory that is utilized by our proposed solution. The mathematics and algorithms of the proposed solution are explained in Chapter 3, also the previous approach is briefly described there and reasons for halting in its development are presented. In Chapter 4, we first discuss the estimation of a sound library as one of the unobserved variables; then, we present an evaluation scheme and evaluation methods and their algorithms; then, we provide tests of the estimation of labels only and tests of the estimation of labels with amplitudes. In Chapter 5, we summarize the resulted information and propose future work.

The author's publications in this topic are the following references: [62, 63, 64, 65, 66, 67, 68, 69]. The same list is cited after Appendix of the thesis.

Chapter 2

Elements of Theory Used by the Proposed Solution

Recall the introduction and the overview of approaches in the AMT, for the inverse music sequencer problem solution, it is suitable to deal with the approaches using a signal model of superposition that is wrapped into a probabilistic framework. The probabilistic framework allows the incorporation of knowledge from the restrictions in music and prior knowledge of sought library sounds. Therefore, we introduce elements of Bayesian estimation theory in this chapter. Since the variational Bayes methods represent a core part of the proposed solution we shall shortly introduce theory regarding this topic, too. A list of utilized probability distributions was taken from [26] and it is placed in Appendix.

2.1 Bayesian Methods

In music signal processing, we are concerned with data $\boldsymbol{D}$ and how we can *infer* a description of a source or a system that generated $\boldsymbol{D}$. The description is represented by a set of unknown parameters (variables) $\boldsymbol{\theta}$ – a vector. In deterministic problems, the inference can be expressed by a rule $\boldsymbol{D} = g(\boldsymbol{\theta})$, in the point of view of superposition of sounds this rule can be interpreted as a signal model. This rule holds for very few data contexts; in most cases, like the superposition of music sounds is, we have to model the uncertainty of the process.

2.1.1 Foundations

The modeling of the uncertainty of a process is described in the theory of probability [70]. In terms of [70], the theory of probability is built on a *decision problem*. The decision problem is characterized by the elements $(\mathcal{E}, \mathcal{C}, \mathcal{A}, \leq)$, where (i) $\mathcal{E}$ is an algebra of relevant events; (ii) $\mathcal{C}$ is a set of possible consequences; (iii) $\mathcal{A}$ is a set of options which consist of functions mapping partitions of the certain event Ω in $\mathcal{E}$ to compatibly-dimensioned ordered sets of elements of $\mathcal{C}$; $\leq$ defining a preference order between some of the elements of $\mathcal{A}$. The need to measure the uncertainty can be satisfied by the definition of the *degree of belief* attached to some event from $\mathcal{E}$. The degree of belief definition is built on coherence (for preventing undesirable implications in comparison of the options $\mathcal{A}$) and quantification (for comparing options in $\mathcal{A}$ and by extension the events and consequences) principles which are designed in a set of axioms. The measure of the degree of belief is the *probability* by the definition [70]. The probability distribution is the sequence of the probabilities over the partition of the certain event Ω.

Having defined the probability and the distribution we can justify the characterization of the probability independence, i.e., if $E, D \in \mathcal{E}$: E being independent of D, then the probability of their intersection is equal to $P(E) \cdot P(D)$. Furthermore, in order to proceed, the conditional probability $P(E|D)$ as the conditional measure of the degree of belief of the event E given D has to be defined. Then, for any $F \neq \emptyset$, we can formulate the conditional probability theorem

$$P(E|D) = \frac{P(E \cap D)}{P(D)}. \tag{2.1}$$

Having expressed (2.1), we can state the Bayes' theorem for any finite partition $\{E_j, j \in J\}$ of Ω and $D \neq \emptyset$:

$$P(E_i|D) = \frac{P(D|E_i) \cdot P(E_i)}{P(D)} \tag{2.2}$$

where $P(D) = \sum_{j \in J} P(D|E_j) \cdot P(E_j)$ which follows from the fact that $D = \cup_j (D \cap E_j)$. If we regard the event D as a relevant piece of evidence, i.e., the data, and the events E_i corresponding to a set of hypotheses about some aspect of the world, then the individual terms of the Bayes' formula are called: $P(E_i|D)$ – a posteriori probability of the E_i, $P(D|E_i)$ – likelihood of the E_i given D, $P(E_i)$ – a priori (prior) probability of the E_i, $P(D)$ – predictive probability. This is the load of the practice – we get a result from an observational experiment that is contained in the data D and our task is to infer the description E_i which generated the data.

Yet, the preference $\leq$ operator from the decision problem definition has been discussed in terms of the events E_j. It follows from the axioms of coherence and quantification [70] that it can be applied on the consequences $\mathcal{C}$. Thus, we can assign a price to each consequence c just as we defined the probability on the events. The price is attached using the *utility function* $u(c) = u(c|c_-, c_+)$ where c_-, c_+ is the worst and the best consequence (for elimination of the pathological, mathematically motivated choices of $\mathcal{C}$), respectively. It remains now to investigate how an overall numerical measure can be assigned to an option $a \in \mathcal{A}$ which reflects the knowledge from both the events of a finite partition of a certain event Ω and from the particular consequences to which these events lead. The measure is called the *expected utility* $\bar{u}$:

$$\bar{u}(a|c_-, c_+, D) = \sum_{j \in J} u(c_j|c_-, c_+) \cdot P(E_j|D) \tag{2.3}$$

where $a \equiv \{c_j|E_j, j \in J\}$ and $D \neq \emptyset$.

2.1.2 Generalizations

We have operated in the area of finite partitions of the certain event Ω. Such setting is not satisfactory because the distribution does not have to be discrete, moreover, for further mathematical operations we need to be able to make the distributions independent of some set of events, therefore we need to be able to integrate. The decision problem is imposed into the *probability space* characterized by $\{\Omega, \mathcal{F}, \mathcal{P}\}$ where $\mathcal{F}$ is σ-algebra of Ω and $\mathcal{P}$ is a complete, σ-additive probability measure on $\mathcal{F}$ [70]. Then we can define the *random quantity* as a function $x : \quad \Omega \to X \subseteqq \mathbb{R}$ (i.e., mapping of the events into $\mathbb{R}$) such that $x^{-1}(B) \in \mathcal{F}, \quad B$ is a Borel set; *cumulative (distribution) function* summing all random quantities in a range yielding the real number in the interval $[0, 1]$; *probability (density) function* $p(\theta)$ being formally defined as a differentiate of the cumulative function integrated over the Borel set (see [70], page 111]); *expectation* of y, $\mathsf{E}(y) = \mathsf{E}(g(x))$, where x, y are random quantities, as the integral (or the sum – discrete case) of $y \cdot p(y)$ over $y \in Y$. Having these terms defined the operation of, e.g., *marginalization* over its complement vector $\boldsymbol{\theta}_2$ is allowed:

$$p(\boldsymbol{\theta}_1|\boldsymbol{D}) \propto \int_{\boldsymbol{\theta}_2^*} p(\boldsymbol{\theta}|\boldsymbol{D}) d\boldsymbol{\theta}_2. \tag{2.4}$$

The revision of the Bayes' theorem yields:

$$p(\boldsymbol{\theta}|\boldsymbol{D}) = \frac{p(\boldsymbol{D}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\boldsymbol{D}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}) d\boldsymbol{\theta}}, \tag{2.5}$$

$$p(\boldsymbol{\theta}|\boldsymbol{D}) \propto p(\boldsymbol{D}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}). \tag{2.6}$$

Now, we can determine the posterior of the random quantity (discrete or continuous) in the form of a vector $\boldsymbol{\theta}$ representing some knowledge of a music signal if we have a music signal data in the matrix $\boldsymbol{D}$ and we are aware of the likelihood of $\boldsymbol{\theta}$, $p(\boldsymbol{D}|\boldsymbol{\theta})$, and the prior knowledge of $\boldsymbol{\theta}$, $p(\boldsymbol{\theta})$. The Bayes' rule can be rewritten in the form of (2.6) because the normalizing constant $\frac{1}{p(\boldsymbol{D})} = \frac{1}{\int_{\boldsymbol{\theta}} p(\boldsymbol{D}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}) d\boldsymbol{\theta}}$ can be usually omitted since it does not depend on $\boldsymbol{\theta}$ moreover it is usually not tractable. After generalization, the probability of intersection of two events (2.1) corresponds to the *joint distribution* of two quantities: $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2) \cdot p(\boldsymbol{\theta}_2)$.

The important quantities which can be asserted about the random quantity $\boldsymbol{\theta}$ are, e.g.: (i) $\mathsf{E}_{p(\boldsymbol{\theta})}(\boldsymbol{\theta})$) – the *mean* of the distribution of the random quantity $\boldsymbol{\theta}$; (ii) $\mathsf{E}_{p(\boldsymbol{\theta})}(\boldsymbol{\theta}^k)$) – the k-th *non-central (absolute) moment*; (iii) $VAR(\boldsymbol{\theta}) = \mathsf{E}_{p(\boldsymbol{\theta})}((\boldsymbol{\theta} - \mathsf{E}_{p(\boldsymbol{\theta})}(\boldsymbol{\theta}))^2)$ – the *variance* (second central moment); $DEV(\boldsymbol{\theta}) = VAR(\boldsymbol{\theta})^{1/2}$ – the *standard deviation; *$M(\boldsymbol{\theta})$ – a *mode* of the distribution of $\boldsymbol{\theta}$, such that $p(M(\boldsymbol{\theta})) = \sup_{\boldsymbol{\theta} \in \boldsymbol{\theta}^*} p(\boldsymbol{\theta})$.

2.2 Inference As the Case of Desicion Problem

With slightly revised notation and terminology, we recall the elements from Subsections 2.1.1, 2.1.2 of the decision problem. The decision problem in the inference context [70] is defined as (i) $\boldsymbol{a} \in \mathcal{A}$ – available "answers" to the inference problem; (ii) $\boldsymbol{\omega} \in \Omega$ – the unknown states of the world (e.g., functions or realizations of a random quantity); (iii) $u : \mathcal{A} \times \Omega \to \mathbb{R}$ – a function assigning utilities to each consequence $(\boldsymbol{a}, \boldsymbol{\omega})$ of a decision to report of an inference of an "answer" $\boldsymbol{a}$ and the state of the world $\boldsymbol{\omega}$; (iv) $p(\boldsymbol{\omega})$ – probability distribution of the state of the world.

The optimal choice of answer to an inference problem is an $\boldsymbol{a}_{opt} \in \mathcal{A}$ which maximizes the expected utility:

$$\boldsymbol{a}_{opt} = \mathrm{argmax}_{\boldsymbol{a} \in \mathcal{A}} \int_{\Omega} u(\boldsymbol{a}, \boldsymbol{\omega}) p(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{2.7}$$

This operation is known as the *decision making*. In literature, statisticians work usually with a so-called *loss function* being defined as $l(\boldsymbol{a}, \boldsymbol{\omega}) = f(\boldsymbol{\omega}) - u(\boldsymbol{a}, \boldsymbol{\omega})$ where f is an arbitrary, fixed function. The maximization of the

expected utility is equivalent to the minimization of the *expected loss* which is sometimes called the *Bayes risk*

$$\int_{\Omega} l(\boldsymbol{a}, \boldsymbol{\omega}) p(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{2.8}$$

The result of the minimization of the Bayes risk is referred to as the *Bayes estimate*. We refer about a *point estimate* once $\mathcal{A} = \Omega$ and $l(\boldsymbol{a}, \boldsymbol{\omega})$ is defined, thus the Bayes estimate is a special case of the point estimate. Selected statistics to calculate a point estimate from "samples" $\boldsymbol{\omega} \in \Omega$ is called the *estimator*.

The examples of the Bayes estimates, given a specific loss function l, are as follows [70, 71, 72]:

- If the loss function is of the quadratic form $l(\boldsymbol{a}, \boldsymbol{\omega}) = (\boldsymbol{a} - \boldsymbol{\omega})^T \boldsymbol{H} (\boldsymbol{a} - \boldsymbol{\omega})$ and $\boldsymbol{H}^{-1}$ exists, then the Bayes estimate is the mean of $p(\boldsymbol{\omega})$, hence $\boldsymbol{a} = \mathsf{E}(\boldsymbol{\omega})$. The statistics to calculate the mean is known as the minimum mean square error (MMSE) estimator.

- If the loss function is of the form $l(\boldsymbol{a}, \boldsymbol{\omega}) = 1 - \mathbb{I}_{||\boldsymbol{a} - \boldsymbol{\omega}|| < \zeta}(\boldsymbol{\omega})$ where $\mathbb{I}$ is the indicator function and ζ is a small number, then the Bayes estimate is the mode of $p(\boldsymbol{\omega})$, assuming that the mode exists. The statistics to calculate the mode is known as the maximum a posteriori (MAP) estimator in the case when there exists some prior knowledge on $\boldsymbol{\omega} \in \Omega$, otherwise it is called the maximum likelihood (ML) estimator.

2.3 Extension to On-Line Case

Let us have an in-time-dynamic system that is described by t observational data $\boldsymbol{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_t]$ where its τ from-start-consecutive data are denoted by $\boldsymbol{D}_\tau = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_\tau]$, and, by t unknown parameters $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_t]$ where its τ from-start-consecutive parameters are denoted by $\boldsymbol{\Theta}_\tau = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_\tau]$, with $\boldsymbol{\Theta}_0 = \{\}$ and $\boldsymbol{D}_0 = \{\}$ by definition. Once again, the Bayes' rule (2.6) is used to update our knowledge of $\boldsymbol{\Theta}_\tau$ in the light of new data $\boldsymbol{d}_\tau$:

$$p(\boldsymbol{\Theta}_\tau | \boldsymbol{D}_\tau) = p(\boldsymbol{\Theta}_\tau | \boldsymbol{d}_\tau, \boldsymbol{D}_{\tau-1}) p(\boldsymbol{\theta}_\tau | \boldsymbol{D}_{\tau-1}, \boldsymbol{\Theta}_{\tau-1}) p(\boldsymbol{\Theta}_{\tau-1} | \boldsymbol{D}_{\tau-1}). \tag{2.9}$$

In order to get the posterior of $\boldsymbol{\theta}_\tau$ instead of posterior for $\boldsymbol{\Theta}_\tau$, we need to integrate (2.9) over $\boldsymbol{\Theta}_{\tau-1}$. If the integrations need to be carried out numerically, the increasing dimensionality proves prohibitive. Let us simplify the

Algorithm 1 Bayesian Filtering

1. The time update

$$p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_{\tau-1}) \equiv p(\boldsymbol{\theta}_\tau), \quad \tau = 1,$$
$$p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_{\tau-1}) = \int_{\theta^*_{\tau-1}} p(\boldsymbol{\theta}_\tau|\boldsymbol{\theta}_{\tau-1}, \boldsymbol{D}_{\tau-1}) p(\boldsymbol{\theta}_{\tau-1}|\boldsymbol{D}_{\tau-1}) d\boldsymbol{\theta}_\tau \quad \tau = 2, 3, \ldots \tag{2.12}$$

2. The data update:

$$p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_\tau) \propto p(\boldsymbol{d}_\tau|\boldsymbol{\theta}_\tau, \boldsymbol{D}_{\tau-1}) p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_{\tau-1}), \quad \tau = 1, 2, \ldots \tag{2.13}$$

calculation of $\boldsymbol{\theta}_\tau$ posterior $p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_\tau)$ by adopting assumptions of the Markov model. Then the observational model results in:

$$p(\boldsymbol{d}_\tau|\boldsymbol{\Theta}_\tau, \boldsymbol{D}_{\tau-1}) = p(\boldsymbol{d}_\tau|\boldsymbol{\theta}_\tau, \boldsymbol{D}_{\tau-1}), \tag{2.10}$$

i.e., $\boldsymbol{d}_\tau$ is conditionally independent of $\boldsymbol{D}_{\tau-1}$ given $\boldsymbol{\theta}_\tau$, and the evolution model is to be simplified as follows:

$$p(\boldsymbol{\theta}_\tau|\boldsymbol{\Theta}_{\tau-1}, \boldsymbol{D}_{\tau-1}) = p(\boldsymbol{\theta}_\tau|\boldsymbol{\theta}_{\tau-1}). \tag{2.11}$$

Application of (2.10), (2.11) onto calculation of posterior $p(\boldsymbol{\theta}_\tau|\boldsymbol{D}_\tau)$, we obtain the equations (2.12), (2.13), which are the part of Bayesian filtering, see Algorithm 1 published, e.g., in [26].

In the case when (2.10), (2.11) are linear in parameters with Gaussian distributed noise as follows:

$$p(\boldsymbol{\theta}_\tau|\boldsymbol{\theta}_{\tau-1}) = \mathcal{N}(\boldsymbol{A}\boldsymbol{\theta}_{\tau-1}, \boldsymbol{R}_{\boldsymbol{\theta}}), \tag{2.14}$$
$$p(\boldsymbol{d}_\tau|\boldsymbol{\theta}_\tau, \boldsymbol{D}_{\tau-1}) = \mathcal{N}(\boldsymbol{C}\boldsymbol{\theta}_\tau, \boldsymbol{R}_{\boldsymbol{d}}), \tag{2.15}$$

where $\boldsymbol{R}_{\boldsymbol{\theta}}$, $\boldsymbol{R}_{\boldsymbol{d}}$, $\boldsymbol{A}$, $\boldsymbol{C}$ are shaping parameters which must be known in advance, then the posterior (2.13) is Gaussian too. This is since the observational model (2.15) is of Gaussian distribution, for which the conjugate prior is Gaussian (see [70], page 266). Algorithm 1 derived for (2.14), (2.15) is called the *Kalman filter* after its inventor. Prediction and update equations for the Kalman filter are presented, e.g., in [73].

In the case when

$$p(\boldsymbol{\theta}_\tau|\boldsymbol{\theta}_{\tau-1}) = \mathcal{N}(a(\boldsymbol{\theta}_{\tau-1}), \boldsymbol{R}_{\boldsymbol{\theta}}), \tag{2.16}$$
$$p(\boldsymbol{d}_\tau|\boldsymbol{\theta}_\tau, \boldsymbol{D}_{\tau-1}) = \mathcal{N}(c(\boldsymbol{\theta}_\tau), \boldsymbol{R}_{\boldsymbol{d}}), \tag{2.17}$$

and a and c are non-linear functions then by using Taylor series the matrices $\boldsymbol{A}$ and $\boldsymbol{C}$ may be approximated by

$$\boldsymbol{A} \approx \frac{\partial a}{\partial \boldsymbol{\theta}_\tau}(\hat{\boldsymbol{\theta}}_\tau), \quad \boldsymbol{C} \approx \frac{\partial c}{\partial \boldsymbol{\theta}_{\tau-1}}(\hat{\boldsymbol{\theta}}_{\tau-1}). \tag{2.18}$$

Such modified Kalman filter is called the extended Kalman filter [73].

2.4 Distributional Approximation

There is a limited number of distributions which allow normalization of Bayes' rule (2.5), marginalization (2.4) and evaluation of moments of posterior distributions. The tractability issues can be bypassed using the numerical integration or using a distribution that approximates the true posterior distribution. The numerical integration is often computationally expensive in higher dimensions, thus an applicable solution can be provided by the distributional approximation

$$p(\boldsymbol{\theta}|\boldsymbol{D}) \approx \tilde{p}(\boldsymbol{\theta}|\boldsymbol{D}). \tag{2.19}$$

According to [74] we may discern

> *Deterministic distributional approximations*: the approximated distribution $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})$ is obtained from $p(\boldsymbol{\theta}|\boldsymbol{D})$ by a technique without any randomness. To the deterministic methods belong: (i) point-based approximation (e.g., gradient search methods, genetic algorithms or neural networks), (ii) Laplace approximation [75], (iii) maximum entropy approximation [76], (iv) variational Bayes free form approximation – our subject of interest.
>
> *Stochastic distributional approximations*: the distribution is approximated using random samples from $p(\boldsymbol{\theta}|\boldsymbol{D})$:
>
> $$\begin{aligned} \boldsymbol{\theta}^{(i)} &\sim p(\boldsymbol{\theta}|\boldsymbol{D}), \\ \{\boldsymbol{\theta}\}_n &= \left\{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(n)}\right\}. \end{aligned} \tag{2.20}$$
>
> Here the effort is to get the smallest number of random samples the best reflecting $p(\boldsymbol{\theta}|\boldsymbol{D})$. A repeated application of random sample sets often leads to an improvement in approximation of $p(\boldsymbol{\theta}|\boldsymbol{D})$. Having the random samples (2.20), the distribution can be written as follows:
>
> $$\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D}) = \frac{1}{n}\sum_{i=1}^{n} \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}), \tag{2.21}$$

where $\delta(\cdot)$ denotes the Dirac δ-function located at $\boldsymbol{\theta}^{(i)}$:

$$\int_{\mathbb{X}} \delta(x - x_0) g(x) dx = g(x_0) \tag{2.22}$$

for the case if $x \in \mathbb{X}$ is a continuous variable and the Kronecker function at $\boldsymbol{\theta}^{(i)}$

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0, \\ 0, & \text{otherwise,} \end{cases} \tag{2.23}$$

if x is a discrete variable. Therefore the posterior moments of $p(\boldsymbol{\theta}|\boldsymbol{D})$ under the empirical approximation (2.21) are

$$\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})}[g(\boldsymbol{\theta})] = \frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{\theta}^{(i)}). \tag{2.24}$$

For low dimensional $\boldsymbol{\theta}$, a temporary set can be generated from a uniform distribution and used in one of standard sampling methods [77] to obtain the representative random samples $\{\boldsymbol{\theta}\}_n$. Retrieval of the representative samples (2.20) for high dimensional distributions is more difficult. To the methods applicable on sampling from high dimensional distributions belong Monte Carlo Markov Chain (MCMC) methods [1, 22, 78], in the online scenario, the representative algorithm is the Particle Filtering method, see the application for multipitch detection problem [46, 79, 80].

2.5 Variational Bayes as the Deterministic Distributional Approximation

Model equations(s) of many problems can be represented by functional(s). The functional represents a mapping that takes a function p from the set of functions $\mathbb{P}$ and maps it to a value. The term "variational" means that the approach follows how the value of the functional changes in response to infinitesimal changes to the input function [81]. The optimization techniques in variational Bayes are fully deterministic. The goal is to select a distribution $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})$ from the space of tractable distributions $\mathbb{P}_c \subset \mathbb{P}$ that are "close" to the true posterior $p(\boldsymbol{\theta}|\boldsymbol{D}) \in \mathbb{P}$. Let us denote $\breve{p}(\boldsymbol{\theta}|\boldsymbol{D}) \in \mathbb{P}_c$ to be a candidate

for the distribution $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})$. Then the optimal selection of the approximating function reads [26]:

$$\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D}) = \text{argmin}_{\breve{p}\in\mathbb{P}_c}\Delta\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) \tag{2.25}$$

here Δ denotes a *dissimilarity function* that assigns a positive scalar as its value.

The selection of an approximation $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})$ can be seen as a decision making (Section 2.2). Regarding the form of the Bayes risk, we need to represent the dissimilarity function Δ as an integral of either $\breve{p}\cdot l(\breve{p},p)$, or $p\cdot l(p,\breve{p})$ over $\boldsymbol{\theta}$. When the loss function l is in the form of either $\log(p/\breve{p})$ or $\log(\breve{p}/p)$, respectively, the dissimilarity function Δ is equal to the *Kullback-Leibler divergence* [82]. It follows from [83] that the loss function reflecting maximum change in information from the data is logarithmic. The Kullback-Leibler (KL) divergence from $\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})$ to $p(\boldsymbol{\theta}|\boldsymbol{D})$ is defined as:

$$\text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) = \int_{\boldsymbol{\theta}*} p(\boldsymbol{\theta}|\boldsymbol{D})\log\frac{\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})}{p(\boldsymbol{\theta}|\boldsymbol{D})}d\boldsymbol{\theta}. \tag{2.26}$$

The important property of the KL divergence is that the KL divergence is not symmetric thus the order of p and $\breve{p}$ must be held.

2.6 Variational Bayes Method

It was first mentioned by Jordan [84] and MacKay [85]. The VB method is based on the optimization functional (2.25) where Δ is represented by the KL divergence (2.26). It results in an iterative approach in which a marginal distribution estimate $\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D})$ is obtained. It is supposed that the approximated distribution is a joint distribution of independent distributions for each $\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_q$. The optimization of the functional is summarized in Theorem 1 and the method itself in Algorithm 2. Both are published in [26].

Theorem 1. *(The VB Theorem). Let $p(\boldsymbol{\theta}|\boldsymbol{D})$ be the posterior distribution of a multivariate parameter $\boldsymbol{\theta}$. The parameter is partitioned into q subvectors of parameters:*

$$\boldsymbol{\theta} = [\boldsymbol{\theta}_1^T,\boldsymbol{\theta}_2^T,\ldots,\boldsymbol{\theta}_q^T]^T. \tag{2.27}$$

Let $\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})$ be an appropriate distribution restricted to the set of conditionally independent distributions for $\boldsymbol{\theta}_1^T,\boldsymbol{\theta}_2^T,\ldots,\boldsymbol{\theta}_q^T$:

$$\breve{p}(\boldsymbol{\theta}|\boldsymbol{D}) = \breve{p}(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_q|\boldsymbol{D}) = \Pi_{i=1}^q\breve{p}(\boldsymbol{\theta}_i|\boldsymbol{D}). \tag{2.28}$$

Then the minimum of distributional approximation

$$\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D}) = \text{argmin}_{\breve{p}\in\mathbb{P}_c}\text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) \tag{2.29}$$

is reached for

$$\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D}) \propto \exp\left(\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})}[\log(p(\boldsymbol{\theta},\boldsymbol{D}))]\right), \quad i = 1,\ldots,q, \tag{2.30}$$

where $\boldsymbol{\theta}_{/i}$ denotes the complement of $\boldsymbol{\theta}_i$ in $\boldsymbol{\theta}$ and $\tilde{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D}) = \Pi_{i=1,j\neq i}^{q}\tilde{p}(\boldsymbol{\theta}_j|\boldsymbol{D})$. We will refer to $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})$ (2.29) as the VB-approximation and $\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D})$ (2.30) as the VB-marginals. The parameters (statistics) of the VB-marginals (2.30) will be called shaping parameters.

According to [26], the proof of Theorem 1 is based on the reduction of the KL divergence

$$\begin{aligned}&\text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) = \\ &\quad = \int_{\boldsymbol{\theta}*}\breve{p}(\boldsymbol{\theta}_i|\boldsymbol{D})\breve{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})\log\left(\frac{\breve{p}(\boldsymbol{\theta}_i|\boldsymbol{D})\breve{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})}{p(\boldsymbol{\theta}|\boldsymbol{D})}\frac{p(\boldsymbol{D})}{p(\boldsymbol{D})}\right)d\boldsymbol{\theta}.\end{aligned}$$

The crucial step of the proof is a selection of a normalizing constant causing reduction on the form

$$\begin{aligned}\text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) =&\text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||\frac{1}{\zeta_i}\mathsf{E}_{\breve{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})}[\log(p(\boldsymbol{\theta},\boldsymbol{D}))]\right) + \\ &+\log p(\boldsymbol{D}) - \log(\zeta_i) + \gamma_i.\end{aligned}$$

When a local minimum of the right-hand side in $\breve{p}$ is sought by calculation of the right-hand side derivation, the one non-trivial solution for the VB-marginal yields $\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D}) = \breve{p}(\boldsymbol{\theta}_i|\boldsymbol{D}) = \frac{1}{\zeta_i}\mathsf{E}_{\breve{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})}[\log(p(\boldsymbol{\theta},\boldsymbol{D}))]$.

An advantageous assumption of the Theorem 1 is that in (2.30) the joint distribution can be combined from the distributions of the same (e.g., exponential) family resulting in a tractable distribution.

In [81] the way they inferred the VB theorem was different. The formula to optimize was given by

$$\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D}) = \mathcal{L}(p(\boldsymbol{\theta}|\boldsymbol{D})) + \text{KL}\left(\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})||p(\boldsymbol{\theta}|\boldsymbol{D})\right) \tag{2.31}$$

where $\mathcal{L}(p(\boldsymbol{\theta}|\boldsymbol{D})) = \int_{\boldsymbol{\theta}*}\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})\log\frac{p(\boldsymbol{\theta},\boldsymbol{D})}{\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})}d\boldsymbol{\theta}$ is the lower bound. The attention is focused there on the lower bound since the minimization of the KL implies maximization of the lower bound. It is explained there that the lower bound is nothing but the negative KL divergence having the fraction in its

Algorithm 2 Iterative VB (IVB) algorithm. Consider $q = 2$ for simplicity, i.e., $\boldsymbol{\theta} = [\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T]^T$. The recursive algorithm is built on eq. (2.30): n-th refinement of the VB-marginal of $\boldsymbol{\theta}_2$ uses n-th refinement of the VB-marginal of $\boldsymbol{\theta}_1$, and $n+1$-th refinement of the VB-marginal of $\boldsymbol{\theta}_1$ uses n-th refinement of the VB-marginal of $\boldsymbol{\theta}_2$. For $n = 1$ the starting value of $\tilde{p}^{[1]}(\boldsymbol{\theta}_1|\boldsymbol{D})$ can be chosen arbitrarily.

1. Update of the VB-marginal of $\boldsymbol{\theta}_2$ at iteration n:

$$\tilde{p}^{[n]}(\boldsymbol{\theta}_2|\boldsymbol{D}) \propto \int_{\boldsymbol{\theta}_{1*}} \tilde{p}^{[n]}(\boldsymbol{\theta}_1|\boldsymbol{D}) \log p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{D}) d\boldsymbol{\theta}_1. \tag{2.32}$$

2. Update of the VB-marginal of $\boldsymbol{\theta}_1$ at iteration $n+1$:

$$\tilde{p}^{[n+1]}(\boldsymbol{\theta}_1|\boldsymbol{D}) \propto \int_{\boldsymbol{\theta}_{2*}} \tilde{p}^{[n]}(\boldsymbol{\theta}_2|\boldsymbol{D}) \log p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{D}) d\boldsymbol{\theta}_2. \tag{2.33}$$

logarithm with the nominator equal to $\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_{/i}|\boldsymbol{D})}[\ln(p(\boldsymbol{\theta}, \boldsymbol{D}))]$ and the denominator equal to $\log\breve{p}(\boldsymbol{\theta}_i|\boldsymbol{D})$. The minimum of the KL divergence happens when the denominator and nominator equals to each other. Some authors [86, 81, 87] use the non-decreasing value of the lower bound as the convergence quality indicator. The lower bound can be also used for number of Gaussians determinations in a Gaussian mixture model [81].

One could find a similarity to the EM algorithm [49, 81]. The similarity resides in the fact that in one step we update a marginal density (E-step) which is utilized in the following step. Unlike the EM algorithm, in the IVB algorithm (i) the expectation formula from the following step is not maximized; (ii) the optimum values of the (shaping) parameters of the distributions are calculated while in the EM the latent variables are updated and optimized directly.

The KL divergence is not symmetric; in case when KL divergence has an order from p to $\breve{p}$, $\mathrm{KL}\left(p(\boldsymbol{\theta}|\boldsymbol{D})||\breve{p}(\boldsymbol{\theta}|\boldsymbol{D})\right)$, then $\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D})$ is equal to the marginal of $p(\boldsymbol{\theta}|\boldsymbol{D})$, i.e., $p(\boldsymbol{\theta}_i|\boldsymbol{D})$. There are two advantages of the iterative VB algorithm over the calculation of the marginal $p(\boldsymbol{\theta}_i|\boldsymbol{D})$: (i) iterative VB algorithm can be summarized in the VB method procedure (Subsection 2.6.1) in which integral calculations do not have to be necessary and (ii) the calculation of the VB algorithm leads to distributions $\tilde{p}(\boldsymbol{\theta}_i|\boldsymbol{D})$ for all i that avoid regions in which $p(\boldsymbol{\theta}|\boldsymbol{D})$ is small (see an example in [81], page 469).

2.6.1 Procedure of the VB Method

Using one family of distributions in design of the joint density $p(\boldsymbol{\theta}, \boldsymbol{D})$ from (2.30) does not have to ensure the tractability of Algorithm 2. The publications describing the VB approach [81, 84, 85] often do not analyze the tractability except of, e.g., [26]. We present here the VB-method published in [26]:

1. **Choose a Bayesian (probability) model**: The joint density $p(\boldsymbol{\theta}, \boldsymbol{D})$ of the model parameters and observed data is created. This step includes the selection of an observation model $p(\boldsymbol{D}|\boldsymbol{\theta})$ and a prior density $p(\boldsymbol{\theta})$ for its parameters.

2. **Partition the parameters**: Separating $\boldsymbol{\theta}$ into q subvectors (2.27). Usually the number of subvectors is equal to the number of latent variables. For simplicity, we will assume that $q = 2$. Then we have to verify that
$$\log p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{D}) = g(\boldsymbol{\theta}_1, \boldsymbol{D})^T h(\boldsymbol{\theta}_2, \boldsymbol{D}), \tag{2.34}$$
where $g(\boldsymbol{\theta}_1, \boldsymbol{D})$ and $h(\boldsymbol{\theta}_2, \boldsymbol{D})$ are finite-dimensional vectors of the same number of dimensions. Specifically, it must be checked if the logarithm of the joint density can be written as a sum of multiples where their first elements contain no $\boldsymbol{\theta}_2$ elements and their second elements contain no $\boldsymbol{\theta}_1$. If the verification fails then VB method will not be tractable.

3. **Write down the VB-marginals**: Theorem 1 is applied. Since the expectation operator E is linear, it can be propagated on the functions g and h. The VB-marginals yield:
$$\begin{aligned} \tilde{p}(\boldsymbol{\theta}_1|\boldsymbol{D}) &\propto \exp\left(\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_2|\boldsymbol{D})}[\log(p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{D}))]\right) \\ &\propto \exp\left(g(\boldsymbol{\theta}_1, \boldsymbol{D})\widehat{h(\boldsymbol{\theta}_2, \boldsymbol{D})}\right), \end{aligned} \tag{2.35}$$
$$\begin{aligned} \tilde{p}(\boldsymbol{\theta}_2|\boldsymbol{D}) &\propto \exp\left(\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_1|\boldsymbol{D})}[\log(p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{D}))]\right) \\ &\propto \exp\left(\widehat{g(\boldsymbol{\theta}_1, \boldsymbol{D})}h(\boldsymbol{\theta}_2, \boldsymbol{D})\right). \end{aligned} \tag{2.36}$$
The identified expectations are
$$\widehat{g(\boldsymbol{\theta}_1, \boldsymbol{D})} \equiv \mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_1|\boldsymbol{D})}[g(\boldsymbol{\theta}_1, \boldsymbol{D})], \tag{2.37}$$
$$\widehat{h(\boldsymbol{\theta}_2, \boldsymbol{D})} \equiv \mathsf{E}_{\tilde{p}(\boldsymbol{\theta}_2|\boldsymbol{D})}[h(\boldsymbol{\theta}_2, \boldsymbol{D})]. \tag{2.38}$$
These expectations are either known from the previous iterations of the VB method or have to be calculated. It is difficult to forecast the tractability for the expectations, e.g., we can operate with $\mathsf{E}_{\tilde{p}(\boldsymbol{\theta}|\boldsymbol{D})}[\log(\overline{g}(\boldsymbol{\theta}, \boldsymbol{D}))]$.

4. **Identify standard distributional forms**: Recognize the standard parametric distribution forms in equations (2.35) and (2.36). Within (2.36), the expectation $g(\widehat{\boldsymbol{\theta}_1, \boldsymbol{D}})$ is taken as a constant and within (2.35), the expectation $h(\widehat{\boldsymbol{\theta}_2, \boldsymbol{D}})$ is taken as a constant. The constant $g(\widehat{\boldsymbol{\theta}_1, \boldsymbol{D}})$ and $h(\widehat{\boldsymbol{\theta}_2, \boldsymbol{D}})$ is "assigned" by the expectation calculated from (2.35) and (2.36), respectively. The found standard distributional forms are

$$\tilde{p}(\boldsymbol{\theta}_1|\boldsymbol{D}) \equiv p(\boldsymbol{\theta}_1|A = \{a_1, a_2, \ldots, a_M\}),$$
$$\tilde{p}(\boldsymbol{\theta}_2|\boldsymbol{D}) \equiv p(\boldsymbol{\theta}_2|B = \{b_1, b_2, \ldots, b_N\}),$$

where A, B denote two sets of shaping parameters of the corresponding VB-marginals. E.g., the Gaussian distribution has got two shaping parameters – the variance and the mean; gamma distribution has got also two shaping parameters $\{b_1, b_2\}$, but the moments need to be calculated from them (in case of the gamma distribution see Appendix).

5. **Formulate necessary VB-moments**: VB-moments are either calculated from the shaping parameters or they correspond to some of the shaping parameters directly, i.e.:

$$\widehat{g(\boldsymbol{\theta}_1)} = \bar{g}(A), \tag{2.39}$$
$$\widehat{h(\boldsymbol{\theta}_2)} = \bar{h}(B). \tag{2.40}$$

6. **Reduce the VB equations**: Equations (2.39), (2.40) and expressions of the shaping parameters are reduced. Commonly, a set of implicit equations is obtained. In rare cases the reduction removes the implicitness (recursion) and provides a closed form solution. In case there are multiple closed form solutions for one VB-marginal, they must be tested to find out which of them is the global minimizer of (2.29). This was not the case in design of our models.

7. **Run the IVB algorithm**: If the closed form solution does not exist, the evaluation of the VB-moments can be accomplished by applying Algorithm 2. Having calculated the shaping parameters $A^{[n]}$, $B^{[n]}$ of the n-th iteration, the VB-moments are obtained

$$\widehat{g(\boldsymbol{\theta}_1)}^{[n]} = \bar{g}(A^{[n]}), \tag{2.41}$$
$$\widehat{h(\boldsymbol{\theta}_2)}^{[n]} = \bar{h}(B^{[n]}). \tag{2.42}$$

The initial values of the shaping parameters should be chosen carefully, because the careful choice may lead to significant computational savings. The stopping condition is either given by the number of iterations or by the difference between the parameter values calculated in two consecutive iterations. Another option is to use the lower bound as in [81, 87, 86].

8. **Report the VB-marginals**: Report the VB-approximation (2.28) in the form of VB-moments.

Chapter 3

Proposed Solution Based on Variational Bayes Method

This chapter proposes a transcription algorithm using a sound memory bank. It is based on the operation of the inverse music sequencer. Its input is an observed audio music signal and its outputs are IDs of the corresponding sounds, their locations in time, their amplitudes and their truncation parameters for each presence of a library sound.

In order to decrease the space of unknown variables and thus number of possible solutions, we will consider that one library sound at a time τ can be present just once, and one library sound will have its allotted amplitude, that is, in our base model it will not be possible to have two presences of a sound with two different amplitudes. In the base model, the missing property of two different amplitude detections for one sound can be managed by generating a sound in various loudnesses and inserting them into the library as one sound.

A music sequencer (not the inverse version) composes the output from sounds stored in the library of sounds $\boldsymbol{\mathcal{F}} = \{\boldsymbol{F}_1, \boldsymbol{F}_2, \dots \boldsymbol{F}_S\}$. Each sound $\boldsymbol{F}_s$ is composed of k_s segments – frames which are supposed to be played one after each other. Its input parameters are defined by: (i) index of the sound to play, $s \in [1, \dots, S]$, (ii) truncation of the sound, i.e., the range of frames from the s-th sound to play, $\boldsymbol{p}_s$, and (iii) amplitude of the sound a_s, $0 \leq a_s \leq 1$. We claim that each sound can be played only once at a time τ. The output sound is obtained by a linear superposition

$$\boldsymbol{y}_\tau = \sum_{s \in \mathcal{S}_\tau} a_s f(\boldsymbol{\mathcal{F}}, s, \boldsymbol{p}_s, \tau), \tag{3.1}$$

where $\boldsymbol{y}_\tau$ is the ϕ-dimensional vector of measurements at time τ that is composed of either time- or frequency-representations of the input music

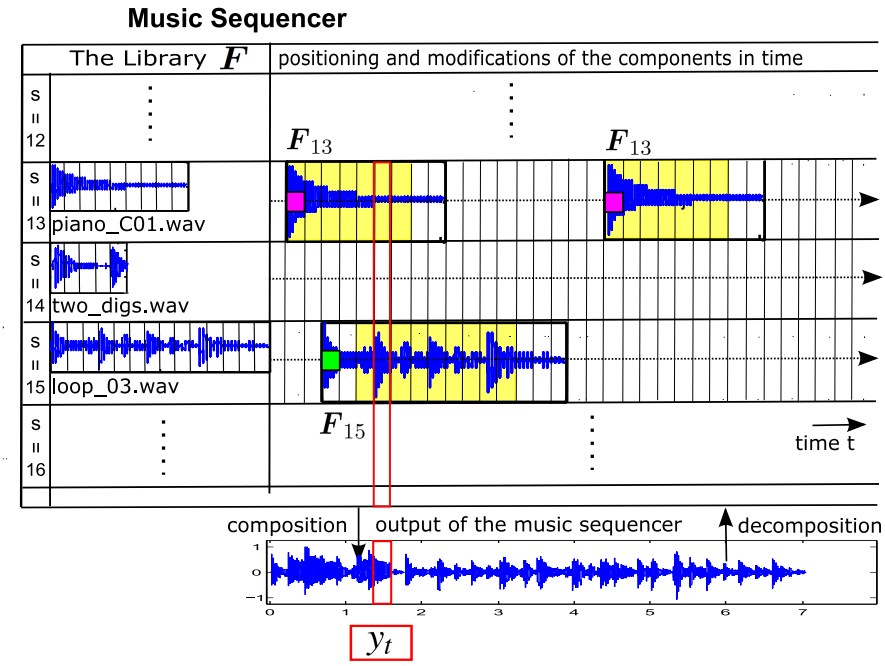


Figure 3.1: Operation of a music and inverse music sequencer. The range of active frames is yellowed. Note that the amplitudes are same for all events in a track s (represented by squares of the same color).

signal segment (frame); s denotes the index of the sound from the set of sounds *active*[1] at time τ, $\mathcal{S}_\tau \subset [1, \ldots, S]$. Function $f(\boldsymbol{\mathcal{F}}, s, \boldsymbol{p}_s, \tau)$ looks up the frame from range $\boldsymbol{p}_s$ of the s-th sound that is active in time τ, see illustration in Fig. 3.1.

3.1 Mathematical model

3.1.1 State Space Model

The recorded signal $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t]$ is modeled by a signal model:

$$\boldsymbol{y}_\tau = \sum_{s \in S} a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau} + \boldsymbol{e}_\tau. \tag{3.2}$$

Here, $\boldsymbol{y}_\tau$ is the ϕ-dimensional vector of measurements at time τ composed of frequency-representation of the input music signal segment (frame).

[1] "Active" implies that their loudness is above a threshold. Sounds of loudness under this threshold are considered to be a silence.

The vector of frequency representation is obtained by the STFT [8] and corresponds to absolute values of one window of the STFT transform, i.e., to a magnitude spectrum; the observations are corrupted with noise $\boldsymbol{e}_\tau$ of Gaussian distribution with zero mean and known covariance matrix $\omega^{-1}\boldsymbol{I}_\phi$; the label process $\boldsymbol{l}_{s,\tau} = [0, 0, \ldots, 1, 0 \ldots, 0]^T$ denotes which frame of the sound is active at time τ, the value one at the first position of the vector $\boldsymbol{l}_{s,\tau} = [1, 0, \ldots]^T$ encodes that the sound s is represented by a silence; $\boldsymbol{\mathcal{F}} = \{\boldsymbol{F}_1, \boldsymbol{F}_2, \ldots \boldsymbol{F}_S\}$ is a library of S pre-recorded sounds; the sound matrix $\boldsymbol{F}_s = [\boldsymbol{f}_{s,1}, \boldsymbol{f}_{s,2}, \ldots, \boldsymbol{f}_{s,k_s+1}]$ is a collection of temporal sequences that consists of the STFT magnitude spectral vectors of the isolated library sound $\boldsymbol{F}_s$. The column vectors of $\boldsymbol{Y}$ and $\boldsymbol{F}_s$ are referred as frames. Since the one at the first position of $\boldsymbol{l}_{s,\tau}$ is to indicate the silence, $\boldsymbol{f}_{s,1}$ is composed of zeros and the first active frame of a sound s starts by $\boldsymbol{f}_{s,2}$, each of the vectors $\boldsymbol{f}_{s,2}, \ldots, \boldsymbol{f}_{s,k_s+1}$ is non-zero; a_s is an amplitude of the sound s which is assumed to be constant in time[2].

Remark. It could be argued that using magnitude STFT spectra in (3.2) is not correct since: when it holds for the original time domain audio data: $time(\boldsymbol{y}_\tau) = time(\boldsymbol{F}_{s_1}) + time(\boldsymbol{F}_{s_2})$ then the superposition principle does not hold for the expectation value of its spectral form[3] [1]:

$$\mathsf{E}(\boldsymbol{y}_\tau) \leq \mathsf{E}(\boldsymbol{F}_{s_1}) + \mathsf{E}(\boldsymbol{F}_{s_2}). \tag{3.3}$$

However, there are authors using it [39, 43], and so do we. The inequality is compensated in amplitudes a_s that are our subject to estimate and are assessed lower than their ground truth.

The sequence of frames within each library sound is ordered in a way that the first frame is typically played first, followed by the second, etc. Formally, the ordering of frames within a sound form a Markov chain

$$\begin{aligned}
\boldsymbol{l}_{s,\tau} &= \boldsymbol{T}\boldsymbol{l}_{s,\tau-1}, && (3.4)\\
\boldsymbol{l}_{s,\tau} &= [1, 0, 0, \ldots, 0, \ldots, 0], && (3.5)\\
\boldsymbol{l}_{s,\tau+1} &= [0, 1, 0, \ldots, 0, \ldots, 0],\\
\boldsymbol{l}_{s,\tau+2} &= [0, 0, 1, \ldots, 0, \ldots, 0],\\
\boldsymbol{l}_{s,\tau+M_{\boldsymbol{p}_s}} &= [0, \underbrace{0, 0, \ldots, 1}_{\boldsymbol{p}_s}, \ldots, 0],\\
\boldsymbol{l}_{s,\tau+M_{\boldsymbol{p}_s}+1} &= [1, 0, 0, \ldots, 0, \ldots, 0],
\end{aligned}$$

where the length of the sequence of active frames $\boldsymbol{p}_s$ is denoted by $M_{\boldsymbol{p}_s}$. The transition matrix $\boldsymbol{T}$ is assumed to be known and of the same form for each

[2] This will be relaxed in Section 3.4.

[3] However it holds for the power spectra: $\mathsf{E}(\boldsymbol{y}_\tau^2) = \mathsf{E}(\boldsymbol{F}_{s_1}^2) + \mathsf{E}(\boldsymbol{F}_{s_2}^2)$ [1].

sound:

$$\boldsymbol{T} = \begin{bmatrix} t_{sil} & t_{end} & t_{end} & \dots & t_{end} \\ t_{start} & \pm & & & \\ t_{start} & t_{next} & \pm & & \\ \vdots & \pm & t_{next} & \pm & \\ t_{start} & & \pm & t_{next} & \pm \end{bmatrix}, \tag{3.6}$$

where t_{end} denotes probability of transition from non-silent to the silent state, t_{start} denotes probability of transition from silence to non-silence, t_{next} is the probability of continuation of the sound to the next frame and t_{sil} the probability of transition from the silent back to itself.

3.1.2 Unobserved Variables

Regarding the observation model (3.2) we consider a_s, $\boldsymbol{F}_s$, $\boldsymbol{l}_{s,\tau}$, ω, for $s = 1 \dots S$, $\tau = 1 \dots t$ to be unobserved variables. The observed data $\boldsymbol{Y}$ is not sufficient since the space of the unobserved variables is large. Therefore additional regularization is imposed on a_s (see in Section 3.2) and $\boldsymbol{F}_s$ (see in Section 3.3). Moreover, our aim is to identify the library sounds with the sources in the observed signal. In the tests of the model, we investigate the sensitivity analysis regarding the unobserved variables. The tests are performed on the case when the unobserved variable is fixed to its ground truth and when it is estimated.

3.1.3 Formulation of Observation Model by Approximation of Poisson Distribution

We try to express the signal model (3.2) in terms of a Poisson distribution which forms the likelihood of our problem for the labels $\boldsymbol{l}_{s,\tau}$, the library sounds $\boldsymbol{F}_s$ and the amplitudes a_s:

$$\boldsymbol{y}_\tau \sim \mathcal{P}o(\boldsymbol{y}_\tau, \sum_s a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau}). \tag{3.7}$$

The distribution model (3.7) can be approximated by a Gaussian distribution model

$$\begin{aligned} &\underbrace{diag\Big((\boldsymbol{Y}\boldsymbol{1}_t \cdot \frac{1}{t})^{-\frac{1}{2}}\Big)}_{\boldsymbol{D}_\phi} \boldsymbol{Y} \underbrace{diag\Big((\boldsymbol{Y}^T\boldsymbol{1}_\phi \cdot \frac{1}{t})^{-\frac{1}{2}}\Big)}_{\boldsymbol{D}_t} = \\ =&diag\Big((\boldsymbol{Y}\boldsymbol{1}_t \cdot \frac{1}{t})^{-\frac{1}{2}}\Big)\Big(\boldsymbol{F}\boldsymbol{A}\boldsymbol{L}\Big)diag\Big((\boldsymbol{Y}^T\boldsymbol{1}_\phi \cdot \frac{1}{t})^{-\frac{1}{2}}\Big) + \tilde{\boldsymbol{E}}, \end{aligned} \tag{3.8}$$

where $\mathbf{1}_t, \mathbf{1}_\phi$ are vectors of ones of size t and ϕ, respectively. The term $\frac{1}{t}$ represents a normalization factor. All library sounds here are denoted by the matrix $\boldsymbol{F} = [\boldsymbol{F}_1, \ldots, \boldsymbol{F}_S]$, the amplitudes are represented by the matrix $\boldsymbol{A} = diag([a_1, a_1, \ldots, a_1, a_2, a_2, \ldots, a_2, \ldots, a_S, a_S, \ldots, a_S])$, where the number of a_s on the diagonal is equal to k_s+1. The term $+1$ is because of the silence. All label vectors of all times τ are combined to the matrix

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{l}_{1,1:t} \\ \boldsymbol{l}_{2,1:t} \\ \vdots \\ \boldsymbol{l}_{S,1:t} \end{bmatrix}. \tag{3.9}$$

The label matrix $\boldsymbol{L}$ and amplitudes $\boldsymbol{a}$ are depicted in Fig. 3.2.

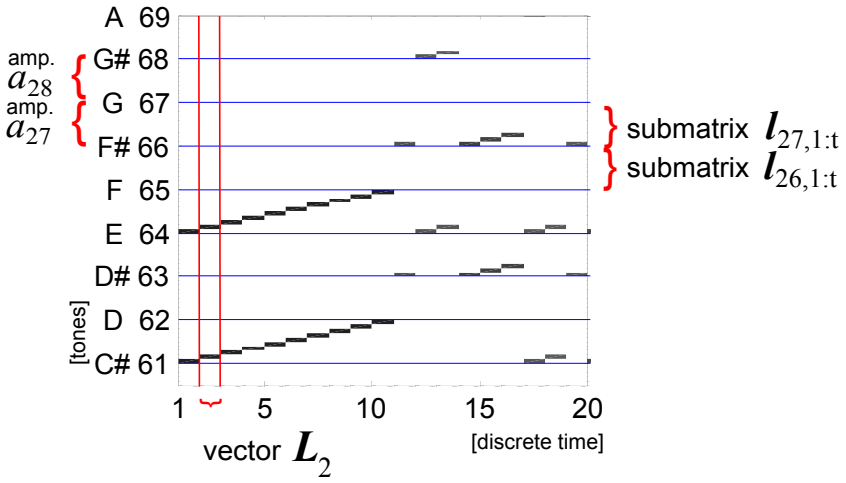


Figure 3.2: Label matrix $\boldsymbol{L}$ and amplitudes $\boldsymbol{a}$ when $k_s = 10$ for all sounds s. The blue lines in the figure denote limits corresponding to one sound. The depiction of $\boldsymbol{L}$ is flipped upside down, therefore the frame sequences run upwards instead of downwards. Such convention will be used in all following displays of $\boldsymbol{L}$. The silence frames are omitted in the picture.

Linear transformation on $\boldsymbol{Y}$ and $\boldsymbol{FAL}$ in (3.8) corresponds to *scaling* in order to whiten the observation noise $\tilde{\boldsymbol{e}}_{1:t} = [\tilde{\boldsymbol{e}}_1, \ldots, \tilde{\boldsymbol{e}}_t]$. We use the scaling from the physiological model of medical image sequences in [26] with an extra term $\frac{1}{t}$ that represents the normalization over length of time. The scaling vector $diag(\boldsymbol{D}_\phi)$ is depicted in Fig. 4.6, 4.7, 4.8, 4.9. Minimization of $d([\boldsymbol{Y}]_{f,\tau}, [\boldsymbol{FAL}]_{f,\tau})$ leads to a ML estimator when the observations are generated by a Poisson process with mean value $[\boldsymbol{FAL}]_{f,\tau}$ [1, 39]. Here d

represents the divergence defined in (1.5). The algorithm for the divergence (1.5) minimization and parameter estimation, when used on the audio data, is the NMF, see in Chapter 1.

In [43], Abdallah and Plumbley investigated an approximation of audio data distribution by the generalized exponential density

$$p(y_f) = \frac{w_f \exp -(w_f y_f)^{\alpha_f}}{\alpha_f^{-1}\Gamma(\alpha_f^{-1})}, \quad y_f > 0, \tag{3.10}$$

where Γ is the gamma function $\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx$. The audio data y_f were the magnitude spectral bins as in our model. They tried to fit the distribution parameters $w_f > 0$, $\alpha_f > 0$ to the audio data using the maximum likelihood criterion. The f indices of w_f values correspond to the elements of the scaling vector $diag(\boldsymbol{D}_\phi)$ in our model. They concluded, that w_f is statistically low for the low frequency bins and high for the high frequency bins ranging between $[10^0, 10^4]$. This is the same conclusion that can be seen from the elements of $diag(\boldsymbol{D}_\phi)$ in Fig. 4.7 – 4.9. In [43], the values of α_f were estimated oscillating around 0.5 . In our model, we use the Gaussian distribution, that is, $\alpha_f = 2$.

The equation (3.8) can be reduced so that operations of $\boldsymbol{D}_\phi$ can be performed on $\boldsymbol{y}_\tau, \boldsymbol{F}_s$ as a pre-processing step which benefits further derivations. When the scaling in spectra is termed $\boldsymbol{D}_\phi$, we have

$$\begin{aligned} p(\tilde{\boldsymbol{e}}_\tau) &\propto \exp(-\frac{1}{2}\omega(\boldsymbol{y}_\tau^{sc} - \sum_s a_s \boldsymbol{F}_s^{sc}\boldsymbol{l}_{s,\tau})^T(\boldsymbol{y}_\tau^{sc} - \sum_s a_s \boldsymbol{F}_s^{sc}\boldsymbol{l}_{s,\tau})), \quad (3.11) \\ \boldsymbol{y}_\tau^{sc} &= \boldsymbol{D}_\phi \boldsymbol{y}_\tau, \\ \boldsymbol{F}_s^{sc} &= \boldsymbol{D}_\phi \boldsymbol{F}_s. \end{aligned}$$

Introducing the scaling in time $\boldsymbol{D}_t$ can be carried out for each τ separately by introducing time varying variance $VAR(\bar{\boldsymbol{e}}_\tau) = \omega^{-1}d_\tau^{-2} \cdot \boldsymbol{I}_\phi$, thus the observation distribution is

$$p(\bar{\boldsymbol{e}}_\tau) \propto \exp(-\frac{1}{2}\omega d_\tau^2(\boldsymbol{y}_\tau - \sum_s a_s \boldsymbol{F}_s\boldsymbol{l}_{s,\tau})^T(\boldsymbol{y}_\tau - \sum_s a_s \boldsymbol{F}_s\boldsymbol{l}_{s,\tau})). \tag{3.12}$$

In the following calculations, all $\boldsymbol{y}_\tau$ and $\boldsymbol{F}_s$ will be regarded to be scaled thus they will refer to $\boldsymbol{y}_\tau^{sc}$ and $\boldsymbol{F}_s^{sc}$, respectively. The scalar constant d_τ^2 from (3.12) will be termed as

$$c_\tau = d_\tau^2. \tag{3.13}$$

In this section, we have discussed the likelihood of our problem for $\boldsymbol{l}_{s,\tau}$, $\boldsymbol{F}_s$, a_s. It is the precision ω we do not have a likelihood for, yet. Hence, given the signal model (3.2), the likelihood for the precision ω can be represented by a Gamma distribution

$$p(\boldsymbol{Y}|\omega) \propto \omega^{t\phi} \exp(\sum_{\tau} -\frac{1}{2}\omega c_\tau (\boldsymbol{y}_\tau - \sum_s a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau})^T (\boldsymbol{y}_\tau - \sum_s a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau})). \quad (3.14)$$

In the following equations, the formula (3.14) is taken as a likelihood for $\boldsymbol{l}_{s,\tau}$, $\boldsymbol{F}_s$, a_s because the term $\omega^{t\phi}$ represents a multiplicative constant in these observation models.

3.2 Approximate Bayesian Identification

The task is to estimate posterior probability of the hidden label process $\boldsymbol{L}_\tau = [\boldsymbol{l}_{1,\tau}^T, \boldsymbol{l}_{2,\tau}^T, \dots \boldsymbol{l}_{S,\tau}^T]^T$, i.e., labels of all sounds in the bank, and their corresponding amplitudes $\boldsymbol{a} = [a_1, a_2, \dots, a_S]^T$. Exact Bayesian inference of model (3.8), (3.4), (3.16), (3.17) via (3.15) is computationally intractable since the number of components in the likelihood (3.15) grows with time. Therefore, we propose to use approximate inference based on Variational Bayes approximation [26]. This technique was successfully used for on-line estimation of mixture models [88]. We follow the steps of the VB method presented in Section 2.6.

1. step:

The joint distribution is constructed:

$$p(\boldsymbol{l}_{1,1:t}, \boldsymbol{l}_{2,1:t} \dots \boldsymbol{l}_{S,1:t}, \boldsymbol{a}, \boldsymbol{\mu}_a, \omega, \boldsymbol{F} = \boldsymbol{F}^{est} | \boldsymbol{y}_{1:t}) \propto$$
$$\prod_{\tau=1}^{t} p(\boldsymbol{y}_\tau|\boldsymbol{L}_\tau, \boldsymbol{a}) p(\boldsymbol{L}_\tau|\boldsymbol{L}_{\tau-1}) p(\boldsymbol{L}_0) p(\boldsymbol{a}) p(\boldsymbol{\mu}_a) p(\omega), \quad (3.15)$$

$$p(\boldsymbol{L}_\tau|\boldsymbol{L}_{\tau-1}) = \prod_{s=1}^{S} p(\boldsymbol{l}_{s,\tau}|\boldsymbol{l}_{s,\tau-1}),$$

where subscript $_{1:t}$ denotes a time sequence, e.g., $\boldsymbol{l}_{s,1:t} = [\boldsymbol{l}_{s,1}, \boldsymbol{l}_{s,2}, \dots \boldsymbol{l}_{s,t}]$ and the expression $\boldsymbol{F} = \boldsymbol{F}^{est}$ corresponds to fixing of the sound library matrix variable $\boldsymbol{F}$ on values $\boldsymbol{F}^{est}$. The observation and transition distribution are given in (3.8), (3.14) and (3.4), respectively. Prior distributions $p(\boldsymbol{L}_0)$ and $p(\omega)$ are chosen as non-informative, in this case uniform and gamma,

respectively. The a priori knowledge of amplitudes is held in the following two distributions:

$$p(\boldsymbol{a}) = \mathcal{N}(\mu_{hyp,a,0} \cdot \mathbf{1}, \boldsymbol{\Sigma}_{a,0}), \tag{3.16}$$

$$p(\mu_{hyp,a,0}) = \mathcal{N}(0, \sigma_{\mu,0}). \tag{3.17}$$

By setting of values of the variances $\boldsymbol{\Sigma}_{a,0}$, $\sigma_{\mu,0}$ we can manage either the same-amplitude-for-all-components estimation or arbitrary amplitude estimation with sparse coding or the fixing of amplitudes at one value.

2. step:

Following the methodology, the unobserved variables can be partitioned according to (2.34), because the logarithm of the joint density (3.15)

$$\begin{aligned}\log p(\boldsymbol{L}_{1:t}, \boldsymbol{a}, \mu_{a,0}, \omega, \boldsymbol{y}_{1:t}) \propto\ & t\phi \cdot \log(\omega) - \\ & - \sum_{\tau} \omega c_\tau (\boldsymbol{y}_\tau - \sum_{s} a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau})^T (\boldsymbol{y}_\tau - \sum_{s} a_s \boldsymbol{F}_s \boldsymbol{l}_{s,\tau}) + \\ & + \boldsymbol{l}_{s,\tau} \log \boldsymbol{T}\, \boldsymbol{l}_{s,\tau-1} - (\boldsymbol{a} - \mathbf{1})^T \boldsymbol{\Sigma}_{a,0}^{-1} (\boldsymbol{a} - \mathbf{1}) + \\ & + \sigma_{\mu,0}^{-1} \mu_{hyp,a,0}\end{aligned} \tag{3.18}$$

can be split into a summation of multiplies (corresponding to the dot product(s) in (2.34)).

3. step:

Minimizing Kullback-Leibler divergence between the left and the right hand side of (3.15), we obtain the following set of implicit equations (VB-marginals):

$$p(\boldsymbol{l}_{s,1:t}|\boldsymbol{y}_{1:t}) \propto \exp\left(\mathsf{E}_{\boldsymbol{a},\mu_{a,0},\omega,\boldsymbol{l}_{\sigma,1:t},\sigma=1\ldots S,\sigma\neq s}(\log p(\boldsymbol{L}_{1:t}, \boldsymbol{a}, \mu_{a,0}, \omega, \boldsymbol{y}_{1:t}))\right), \tag{3.19}$$

$$p(\boldsymbol{a}|\boldsymbol{y}_{1:t}) \propto \exp\left(\mathsf{E}_{\mu_{a,0},\omega,\boldsymbol{l}_{\sigma,1:t},\sigma=1\ldots S}(\log p(\boldsymbol{L}_{1:t}, \boldsymbol{a}, \mu_{a,0}, \omega, \boldsymbol{y}_{1:t}))\right), \tag{3.20}$$

$$p(\mu_{a,0}|\boldsymbol{y}_{1:t}) \propto \exp\left(\mathsf{E}_{\boldsymbol{a},\omega,\boldsymbol{l}_{\sigma,1:t},\sigma=1\ldots S}(\log p(\boldsymbol{L}_{1:t}, \boldsymbol{a}, \mu_{a,0}, \omega, \boldsymbol{y}_{1:t}))\right), \tag{3.21}$$

$$p(\omega|\boldsymbol{y}_{1:t}) \propto \exp\left(\mathsf{E}_{\boldsymbol{a},\mu_{a,0},\boldsymbol{l}_{\sigma,1:t},\sigma=1\ldots S}(\log p(\boldsymbol{L}_{1:t}, \boldsymbol{a}, \mu_{a,0}, \omega, \boldsymbol{y}_{1:t}))\right). \tag{3.22}$$

The eq. (3.19) – (3.22) are tractable since their logarithm can be split on summation of multiplies where E is applied only on expressions corresponding to moments.

4. step:

Identification of standard distributional forms: substituting (3.4), (3.5), (3.7), (3.16), (3.17) into (3.19) and making some necessary simplifications (see Appendix) we obtain:

$$p(\boldsymbol{l}_{s,1:t}|\boldsymbol{y}_t) \propto \exp\Big(\mathsf{E}_{\boldsymbol{a},\mu_{a,0},\omega,\boldsymbol{l}_{\sigma,1:t},\sigma=1\ldots S,\sigma\neq s}\Big(\sum_{\tau=1}^{t} -\frac{1}{2}c_\tau\omega(\boldsymbol{y}_\tau - \sum_s a_s\boldsymbol{F}_s\boldsymbol{l}_{s,\tau})^T(\boldsymbol{y}_\tau - \sum_s a_s\boldsymbol{F}_s\boldsymbol{l}_{s,\tau}) + + \boldsymbol{l}_{s,\tau}\log\boldsymbol{T}\,\boldsymbol{l}_{s,\tau-1}\Big)\Big), \tag{3.23}$$

$$\propto \prod_{\tau=1}^{t}\prod_{i=1}^{dim(F_s)} o_{i,\tau}\boldsymbol{t}_{i,:}\boldsymbol{l}_{s,\tau-1}, \tag{3.24}$$

$$o_{i,\tau} \propto \exp\left(-\frac{1}{2}\hat{\omega}c_\tau(\tilde{\boldsymbol{y}}_\tau - \hat{a}_s\boldsymbol{f}_{s,i})^T(\tilde{\boldsymbol{y}}_\tau - \hat{a}_s\boldsymbol{f}_{s,i})\right)\cdot \cdot\exp\left(-\frac{1}{2}\hat{\omega}c_\tau\Sigma_{a,s,s}\boldsymbol{f}_{s,i}^T\boldsymbol{f}_{s,i}\right), \tag{3.25}$$

$$\tilde{\boldsymbol{y}}_\tau = \boldsymbol{y}_\tau - \sum_{\sigma=1,\sigma\neq s}^{S}\hat{a}_\sigma\boldsymbol{F}_\sigma\hat{\boldsymbol{l}}_{\sigma,\tau}, \tag{3.26}$$

$$p(\boldsymbol{a}|\boldsymbol{y}_{1:t}) = \mathcal{N}(\boldsymbol{\mu}_a,\boldsymbol{\Sigma}_a),$$

$$\boldsymbol{\mu}_a = \boldsymbol{\Sigma}_a\left(\sum_{\tau=1}^{t}\mathsf{E}(\Phi_\tau)^T y_\tau + \sigma_{a,0}^{-1}\hat{\boldsymbol{a}}_{hyp,a,0}\cdot\mathbf{1}\right), \tag{3.27}$$

$$\boldsymbol{\Sigma}_a = \left(\sum_{\tau=1}^{t}\mathsf{E}(\Phi_\tau^T\Phi_\tau) + \sigma_{a,0}^{-1}\right)^{-1}, \tag{3.28}$$

where $\hat{\boldsymbol{a}}_{hyp,a,0}$ and $\hat{\omega}$ are hyperparameters of the model estimated from the marginal distributions

$$
\begin{aligned}
p(\mu_{a,0}|\boldsymbol{y}_{1:t}) &= \mathcal{N}(\mu_{hyp,a,0}, \sigma_{hyp,a,0}), \\
\mu_{hyp,a,0} &= \sigma_{hyp,a,0}\sigma_{a,0}^{-1}(\sum_s \hat{a}_s), \qquad (3.29) \\
\sigma_{hyp,a,0} &= (\sum_s \sigma_{a,0}^{-1} + \sigma_{\mu,0}^{-1})^{-1}, \qquad (3.30) \\
p(\omega|\boldsymbol{y}_{1:t}) &= \mathcal{G}(a, b), \\
a &= t\phi, \\
b &= \sum_{\tau=1}^{t}(\boldsymbol{y}_\tau - \sum_s \hat{a}_s \boldsymbol{F}_s \hat{\boldsymbol{l}}_{s,\tau})^T(\boldsymbol{y}_\tau - \sum_s \hat{a}_s \boldsymbol{F}_s \hat{\boldsymbol{l}}_{s,\tau}) \\
&\quad + \sum_\tau \mathrm{trace}(\sigma_{a,0}\boldsymbol{I} \cdot \mathsf{E}(\Phi_\tau^T \Phi_\tau)). \qquad (3.31)
\end{aligned}
$$

Note that (3.23) can be rewritten as

$$
p(\boldsymbol{l}_{s,1:t}|\boldsymbol{y}_{1:t}) \propto \prod_{\tau=1}^{T} p(\tilde{\boldsymbol{y}}_\tau|\boldsymbol{l}_{s,\tau}) p(\boldsymbol{l}_{s,\tau}|\boldsymbol{l}_{s,\tau-1}) p(\boldsymbol{l}_{s,0}) \qquad (3.32)
$$

which is the standard hidden Markov model that could be solved using the forward-backward algorithm. However, due to dependence of $\tilde{\boldsymbol{y}}_\tau$ on the expected values $\hat{\boldsymbol{l}}_{\sigma,\tau}$, it is used only as a subroutine within Algorithm 3.

5. step, 6. step:

Formulate VB moments and reduce them: the expectations are

$$
\begin{aligned}
\hat{l}_{s,i,\tau} &= \hat{l}_{s,i,\tau}^{nonorm} / \sum_{i=1}^{k_s} \hat{l}_{s,i,\tau}^{nonorm}, \quad \hat{l}_{s,i,\tau}^{nonorm} = o_{i,\tau} \boldsymbol{t}_{i,:} \boldsymbol{l}_{s,\tau-1}, \qquad (3.33) \\
\hat{\boldsymbol{a}} &= \boldsymbol{\mu}_a, \qquad (3.34) \\
\hat{a}_{hyp,a,0} &= \mu_{hyp,a,0}, \qquad (3.35) \\
\mathsf{E}(\Phi_\tau^T \Phi_\tau) &= \hat{\omega} c_\tau [\boldsymbol{F}_1 \hat{\boldsymbol{l}}_{1,\tau}, \ldots, \boldsymbol{F}_S \hat{\boldsymbol{l}}_{S,\tau}]^T [\boldsymbol{F}_1 \hat{\boldsymbol{l}}_{1,\tau}, \ldots, \boldsymbol{F}_S \hat{\boldsymbol{l}}_{S,\tau}], \qquad (3.36) \\
\mathsf{E}(\Phi_\tau) &= \hat{\omega} c_\tau [\boldsymbol{F}_1 \hat{\boldsymbol{l}}_{1,\tau}, \ldots, \boldsymbol{F}_S \hat{\boldsymbol{l}}_{S,\tau}], \qquad (3.37) \\
\hat{\omega} &= \frac{a}{b}. \qquad (3.38)
\end{aligned}
$$

We note that the vector $\hat{\boldsymbol{a}}_{hyp,a,0}$ contains identical elements equal to $\hat{a}_{hyp,a,0}$.

Algorithm 3 Off-line VB algorithm for the proposed solution

1. Set initial conditions: $\hat{\boldsymbol{l}}_{s,\tau}^{(0)} = [0, \frac{1}{k_s}, \frac{1}{k_s}, \frac{1}{k_s}, \ldots]^T, \forall s, \tau$, $\boldsymbol{a}^{(0)} = \mathbf{1}$, set iterative counter $i = 0$, set nuisance parameters $[r_\omega, \sigma_{\mu,0}, \sigma_{a,0}, t_{sil}, t_{end}, t_{start}, t_{next}]$. If $\mu_{hyp,a,0}$ is considered to be estimated, then $\mu_{hyp,a,0}^{(0)} = 1$, else if it is considered to be fixed, then $\mu_{hyp,a,0}^{(0)} = value$.
2. Update statistics of $p(\boldsymbol{a}|\boldsymbol{y}_{1:t})$ using (3.27), (3.28).
3. If $\mu_{hyp,a,0}$ is considered to be estimated, then
   (a) update statistics of $p(\mu_{a,0}|\boldsymbol{y}_{1:t})$ using (3.29),
   (b) else assign the fixed value $\mu_{hyp,a,0}^{(i)} = value$.
4. Update statistics of $p(\omega|\boldsymbol{y}_{1:t})$ according to Algorithm 4.
5. For $s = 1, \ldots, S$
   (a) compute $\tilde{\boldsymbol{y}}_\tau$ using available estimates (3.26),
   (b) evaluate mean $\hat{\boldsymbol{l}}_{s,\tau}^{(i)}$ via forward-backward algorithm using (3.33).
6. $i = i + 1$.
7. If $i < max\_iterations$ and $distance(\hat{\boldsymbol{l}}_{s,\tau}^{(i)}, \hat{\boldsymbol{l}}_{s,\tau}^{(i-1)}) > threshold$ goto 2, end otherwise.

7 – 8. step:

Running the IVB algorithm is accomplished through Algorithm 3 and the variables $\hat{\boldsymbol{L}}$,$\boldsymbol{L}$,$\hat{\boldsymbol{a}}$, $\hat{a}_{hyp,a,0}$, $\hat{\omega}$ are reported. The subvectors $\boldsymbol{l}_{s,\tau}$ of matrix variable $\boldsymbol{L}$ contain one value of one. The position of the one in the label vector $\boldsymbol{l}_{s,\tau} = [0, 0, \ldots, 1, \ldots, 0]^T$ is determined by the maximum value from the elements of $\hat{\boldsymbol{l}}_{s,\tau}$.

The calculation of ω estimate was not done by a simple fill in the formula (3.38). The difference $\boldsymbol{r}_\tau = \boldsymbol{y}_\tau - \sum_s \hat{a}_s \boldsymbol{F}_s \hat{\boldsymbol{l}}_{s,\tau}$ from (3.31) yields low values in uninformative frequency bins too, that is, when both $y_{j,\tau}$ and $f_{s,j,\tau}$ are very low. If the uninformative bins are retained in the calculation of ω estimate, the estimated precision $\hat{\omega}$ results in a high value already in early iterations hence the convergence is stopped prematurely. In Algorithm

Algorithm 4 Calculation of ω estimate

1. Initialize the number of frequency bins that are above the threshold r_ω: $N_{\boldsymbol{r}} = 0$.
2. For $\tau = 1, \ldots, T$,
   (a) Calculate $\hat{\boldsymbol{y}}_\tau = \sum_s \hat{a}_s \boldsymbol{F}_s \hat{\boldsymbol{l}}_{s,\tau}$.
   (b) For $j = 1, \ldots, \phi$,
      i. If $y_{\tau,j} > r_\omega$ or $\hat{y}_{\tau,j} > r_\omega$,
         A. $\tilde{f}_{s,i,j} = f_{s,i,j}$, where $f_{s,i,j}$ is j-th frequency bin of the library sound vector $\boldsymbol{f}_{s,i}$,
         $N_{\boldsymbol{r}} = N_{\boldsymbol{r}} + 1$,
         B. else $\tilde{f}_{s,i,j} = 0$.
3. $\tilde{b} = b(\tilde{\boldsymbol{\mathcal{F}}})$, where $\tilde{\boldsymbol{\mathcal{F}}}$ denotes the library of sounds calculated in A. The modified library $\tilde{\boldsymbol{\mathcal{F}}}$ is used in b calculation (3.38) instead of $\boldsymbol{\mathcal{F}}$.
   $\hat{\omega} = \frac{N_{\boldsymbol{r}}}{\tilde{b}}$.

4, the parameter r_ω represents the threshold of how the frequency bins are informative.

3.3 Extension to Unknown Library of Sounds

When the library sounds $\boldsymbol{F}_s$ are considered as unobserved variables the number of free parameters in the model (3.2) increases rapidly. In order to allow a meaningful estimation of unobserved variables from the model either the observed data $\boldsymbol{Y}$ need to be long enough in time or some additional regularization needs to be imposed on $\boldsymbol{F}$. Let us consider

$$p(\boldsymbol{F}_s) = \mathcal{N}(\boldsymbol{F}_s | \boldsymbol{F}_s^{est}, \boldsymbol{\Sigma}_f), \tag{3.39}$$

where $\boldsymbol{F}_s^{est}$ denotes one sound from the library of sounds for estimation (i.e., from the e-bank, see in Chapter 4) and $\boldsymbol{F}_s$ is a sound which is to be estimated. The covariance matrix $\boldsymbol{\Sigma}_f$ is defined as $\boldsymbol{\Sigma}_f = \xi \boldsymbol{I}$. The coefficient ξ can be understood as a balance term between the blind source separation model ($\xi \to 0$) and a model where $p(\boldsymbol{F}_s)$ depends just on the selection of the

library ($\xi \to \infty$). The latter results in

$$p(\boldsymbol{F}_s) = \delta(\boldsymbol{F}_s - \boldsymbol{F}_s^{est}). \tag{3.40}$$

Regarding the model (3.39) the joint distribution (3.15) is filled in by the prior density $p(\boldsymbol{F}_s)$ – 1. step. Since $p(\boldsymbol{F}_s)$ is of gaussian, the logarithm of the joint density is split-able on functions h, g from Subsection 2.6.1 (2. step).

3. step: If eq. (3.15) is complemented by

$$p(\boldsymbol{F}|\boldsymbol{y}_t) \equiv \prod_{s=1}^{S} p(\boldsymbol{F}_s|\boldsymbol{y}_t) \tag{3.41}$$

on the right hand side and by $p(\boldsymbol{F}_s)$ on the left hand side, minimizing the KL divergence between them we obtain a new VB marginal

$$p(\boldsymbol{F}_s|\boldsymbol{y}_{1:t}) \propto \exp\left(\mathsf{E}_{\boldsymbol{a},\mu_{a,0},\omega,\boldsymbol{l}_{1:S,1:t},\boldsymbol{F}_\sigma,\sigma=1\ldots S,\sigma\neq s}(\ln p(\boldsymbol{L}_{1:t},\boldsymbol{a},\mu_{a,0},\omega,\boldsymbol{F},\boldsymbol{y}_{1:t}))\right). \tag{3.42}$$

The eq. (3.42) is tractable since in case of the library the linear operator E propagation ends at $\mathsf{E}(\boldsymbol{F}^v)$, where v is an exponent. In all other VB marginals (3.19) – (3.22), the operator E must be applied to the variables $\boldsymbol{F}_1, \ldots, \boldsymbol{F}_S$, too.

4. step: Substituting (3.4), (3.5), (3.7), (3.16), (3.17), (3.39) into the extended (3.19) and making some necessary simplifications we identify the following new distributional form with shaping parameters:

$$p(\boldsymbol{F}_s|\boldsymbol{y}_{1:t}) \propto \mathcal{N}(\boldsymbol{F}_s^{\boldsymbol{\mu}}, \boldsymbol{I}_{k_s} \otimes \boldsymbol{\Sigma}_{\boldsymbol{F}_s}) \tag{3.43}$$

$$\boldsymbol{F}_s^{\boldsymbol{\mu}} = \boldsymbol{\Sigma}_{\boldsymbol{F}_s}(\hat{\omega}\hat{a}_s^T \boldsymbol{Y}\hat{\boldsymbol{l}}_{s,1:t}^T + \xi \boldsymbol{F}_s^{est}), \tag{3.44}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{F}_s} = \left(\hat{\omega}(\hat{a}_s^2 + VAR(a_s))\left(\sum_\tau \mathrm{diag}(\hat{\boldsymbol{l}}_{s,\tau})\right) + \xi \boldsymbol{I}_{k_s}\right)^{-1}. \tag{3.45}$$

5., 6. step: The current model has to be filled in by a new formula for diagonal elements of $\mathsf{E}(\Phi_\tau'\Phi_\tau)$ because the operator E has to be applied on $\boldsymbol{F}_s$ in $p(\boldsymbol{a}|\boldsymbol{y}_{1:t})$, $p(\omega|\boldsymbol{y}_{1:t})$. Thus, the set of expectations (3.33) – (3.38) must be filled in by the new expectations:

$$\begin{aligned}
\mathsf{E}(\Phi_\tau^T\Phi_\tau)_{s,s} &= \omega\mathsf{E}(\boldsymbol{l}_{s,\tau}^T \boldsymbol{F}_s^T \boldsymbol{F}_s \boldsymbol{l}_{s,\tau}) = \hat{\omega}\hat{\boldsymbol{l}}_{s,\tau}^T \mathrm{diag}(\mathsf{E}(\boldsymbol{F}_s^T\boldsymbol{F}_s)),\\
\mathsf{E}(\Phi_\tau^T\Phi_\tau)_{s,\sigma} &= \omega\mathsf{E}(\boldsymbol{l}_{s,\tau}^T \boldsymbol{F}_s^T \boldsymbol{F}_\sigma \boldsymbol{l}_{\sigma,\tau}) = \hat{\omega}\hat{\boldsymbol{l}}_{s,\tau}^T \hat{\boldsymbol{F}}_s^T \hat{\boldsymbol{F}}_\sigma \hat{\boldsymbol{l}}_{\sigma,\tau},\\
\mathsf{E}(\boldsymbol{F}_s^T\boldsymbol{F}_s) &= \hat{\boldsymbol{F}}_s^T\hat{\boldsymbol{F}}_s + \boldsymbol{\Sigma}_{\boldsymbol{F}_s},\\
\hat{\boldsymbol{F}}_s^{\boldsymbol{\mu}} &= \boldsymbol{F}_s^{\boldsymbol{\mu}}.
\end{aligned} \tag{3.46}$$

Algorithm 5 Off-line VB algorithm for the proposed solution with library estimation

1. Set initial conditions: The initializations in Algorithm 3 are complemented by initialization for the library of sounds: $\hat{\boldsymbol{F}}_s^{\boldsymbol{\mu}(0)} = \boldsymbol{F}_s^{est}$ for all $s = 1 \dots S$.
2. Identical to Algorithm 3.
3. Identical to Algorithm 3.
4. Identical to Algorithm 3.
5. For $s = 1, \dots, S$
   (a) compute $\tilde{\boldsymbol{y}}_\tau$ using available estimates (3.26),
   (b) evaluate mean $\hat{\boldsymbol{l}}_{s,\tau}^{(i)}$ via forward-backward algorithm using (3.33),
   (c) update statistics of $p(\boldsymbol{F}_s|\boldsymbol{y}_{1:t})$ using (3.44), (3.45).
6. $i = i + 1$.
7. If $i < max\_iterations$ and $distance(\hat{\boldsymbol{l}}_{s,\tau}^{(i)}, \hat{\boldsymbol{l}}_{s,\tau}^{(i-1)}) > threshold$ goto 2, end otherwise.

Algorithm 3 is complemented by estimation of (3.46) resulting in Algorithm 5.

3.4 Extension of the Algorithm to Recursive Estimation

Algorithm 3 can be easily extended to recursive form by running on a moving window of length w. This would allow estimation of amplitudes changing over time. We carry out the Bayesian estimation of labels on the moving window $\boldsymbol{L}_{\tau-w:\tau}$ via the Bayes rule:

$$p(\boldsymbol{L}_{\tau-w:\tau}|\boldsymbol{y}_{1:t}) \propto \prod_{\tau=t-w}^{t} p(\boldsymbol{y}_\tau|\boldsymbol{a}_s, \boldsymbol{L}_\tau) p(\boldsymbol{L}_\tau|\boldsymbol{L}_{\tau-1}) p(\boldsymbol{L}_{\tau-w}) p(\boldsymbol{a}). \tag{3.47}$$

The prior distribution $p(\boldsymbol{L}_{\tau-w})$ in the Bayes rule (3.47) is the delayed posterior and amplitudes $\boldsymbol{a}$ are now considered stationary only with respect to the moving window. The estimation is carried out in Algorithm 6.

Algorithm 6 Recursive VB algorithm

(i) Set initial estimates $\hat{\boldsymbol{L}}_0^{(0)}$,$\boldsymbol{a}^{(0)}$.
(ii) For each time τ do:

- The following operations concern quantities within a window determined by τ. Use estimates from the previous step as initializers, i.e., $\forall s : \hat{\boldsymbol{l}}_{s,\tau-w:\tau}^{(0)} = [\hat{\boldsymbol{l}}_{s,\tau-1-w:\tau-1}^{(i_{\tau-1})}, \hat{\boldsymbol{l}}_{s,\tau}^{(0)}]^T, \hat{a}_{s,\tau}^{(0)} = \hat{a}_{s,\tau-1}^{(i_{\tau-1})}$, set iterative counter $i_\tau = 0$. Set nuisance parameters $[r_\omega, \sigma_{\mu,0}, \sigma_{a,0}, t_{sil}, t_{end}, t_{start}, t_{next}]$. If $\mu_{hyp,a,0}$ is considered to be estimated, then $\mu_{hyp,a,0}^{(0)} = 1$, else if it is considered to be fixed, then $\mu_{hyp,a,0}^{(0)} = value$.

  1. Execute points (2), (3), (4), (5) from Algorithm 3.
  2. $i_\tau = i_\tau + 1$.
  3. If $i < max\_iterations$ and $distance(\hat{\boldsymbol{l}}_{s,\tau}^{(i_\tau)}, \hat{\boldsymbol{l}}_{s,\tau}^{(i_\tau-1)}) > threshold$ goto (a).

The window length w allows to tune properties of the algorithm. In each step and for each sound s, the algorithm estimates w time delayed labels and one amplitude. The delayed labels are improved estimates of previous labels and the amplitude. For window-length $w > 1$, the online algorithm re-iterates estimates of the labels from the previous steps thus, e.g., for $max\_iterations = 1$, the estimates at the discrete times after the window length w are iterated w times. Each window can be processed by a separate sound library which can be prepared in the pre-processing stage, as it is pointed in Section 4.7.

3.5 Properties of Algorithm 3

An important property of the algorithm lies in its strong discrimination among frames of a library sound. This is demonstrated in Fig. 3.3: there we have a visual comparison between two algorithms estimating parameters from the same observation model (3.7), the first is the proposed variational Bayes algorithm that uses the model without the transition part (i.e., $\hat{l}_{s,i,\tau}^{nonorm} = o_{i,\tau}$), the second is the NMF algorithm based on the ML estimate from (3.7).

3.6 Previous Approach Resulting in Extended Kalman Filter Algorithm

In Eusipco 2010 [68], we described an online model. The model did not contain the restriction that only one frame of a library sound can be present at time τ. The amplitude was allowed to move over time. The observational model was given by the following formula

$$p(\boldsymbol{y}_\tau|\boldsymbol{a}_\tau, \boldsymbol{F}) = \mathcal{N}(\boldsymbol{F}\boldsymbol{a}_\tau, \omega^{-1}\boldsymbol{I}_\phi). \tag{3.48}$$

The task was to estimate posterior density of $\boldsymbol{a}_\tau$ given available data, $p(\boldsymbol{a}_\tau|\boldsymbol{F}, \boldsymbol{Y}_\tau)$, where $\boldsymbol{Y}_\tau = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_\tau]$. The constraints on amplitudes were transformed into the Gaussian prior $p(\boldsymbol{a}_\tau|\boldsymbol{a}_{\tau-1})$, which is parametrized by a mean value of size N and covariance matrix of size $N \times N$.

We defined a simple transformation between discrete variable $\boldsymbol{\alpha}_\tau$ and continuous amplitude $\boldsymbol{a}_\tau$, specifically

$$p(a_{i,\tau}|\alpha_{i,\tau}) = \begin{cases} \mathcal{N}(1, k\sigma_1), & \text{if } \alpha_{i,\tau} = 1, \\ \mathcal{N}(0, \sigma_1), & \text{otherwise.} \end{cases} \tag{3.49}$$

Intuitively, zero values of $\alpha_{i,\tau}$ (i.e., representation of silence) were mapped on $a_{i,\tau}$ which are close to zero and $\alpha_{i,\tau} = 1$ were mapped to $a_{i,\tau}$ close to 1. The proximity was modeled by variance parameter σ_1. Since we allowed lower amplitudes of the tone, we modeled the variance of the first component of the distribution in (3.49) to be k times greater than that of the second component. Inverse mapping of $\boldsymbol{a}_\tau$ to $\boldsymbol{\alpha}_\tau$ can be obtained by the Bayes rule:

$$p(\alpha_{i,\tau}|a_{i,\tau}) = p(a_{i,\tau}|\alpha_{i,\tau})p(\alpha_{i,\tau})/p(a_{i,\tau}). \tag{3.50}$$

In the discrete parametrization (3.1), the transition between frames was modeled by a simple Markov transition:

$p(\alpha_{i,\tau}\|\alpha_{i-1,\tau-1})$	$\alpha_{i-1,\tau-1} = 0$	$\alpha_{i-1,\tau-1} = 1$
$\alpha_{i,1} = 0$	ζ_0	$1 - \zeta_0$
$\alpha_{i,1} = 1$	$1 - \zeta_1$	ζ_1

where τ_0, τ_1 are constant probabilities that the discrete amplitude is not changed by the transition from $\tau - 1$ to τ . This transition model can be combined with (3.50) as follows:

$$p(a_{i,\tau}|a_{i-1,\tau-1}) = \sum_{\alpha_{i,\tau-1}} \sum_{\alpha_{i-1,\tau-1}} p(a_{i,\tau}|\alpha_{i,\tau})p(\alpha_{i,\tau}|\alpha_{i-1,\tau-1})p(\alpha_{i-1,\tau-1}|a_{i-1,\tau-1}). \tag{3.51}$$

Direct application of this rule would result in prior $p(\boldsymbol{a}_t)$ being a mixture of 4^{Kt} components which is not computationally tractable. Hence, we projected (3.51) into the single Gaussian density

$$p(a_{i,\tau}|a_{i-1,\tau-1}) = \mathcal{N}(\mu_{i,\tau-1}, \sigma_{i,\tau-1}) \tag{3.52}$$

using geometric merging of probabilities [89]. Since $p(\boldsymbol{a}_\tau|\boldsymbol{a}_{\tau-1})$ was non-linear, the extended Kalman filtering (EKF) algorithm was utilized to estimate $\boldsymbol{a}_\tau$. Moreover, in order to utilize knowledge also after τ to estimate $\boldsymbol{a}_\tau$, the Rauch-Tung-Striebel two-pass smoother [73] was applied. The transformation of discrete $\boldsymbol{\alpha}_\tau$ onto continuous $\boldsymbol{a}_\tau$ in the model considerably increased the accuracy of estimation over simple multiplication of the transitional and sparsity constraints, both given by Gaussians, see in [68].

A visual comparison between the EKF algorithm based on this model and the current VB algorithm can be seen in Fig. 3.3. We stopped the development in the EKF approach because:

- Each prior knowledge on $\boldsymbol{a}_\tau$ needs to be transformed in such a way to get a Gaussian $p(\boldsymbol{a}_\tau|\boldsymbol{a}_{\tau-1})$. This prevents the greater freedom in suitable distribution selection and further improvements. Another approximation is performed using derivatives in the extension of the Kalman filter, see in Section 2.3.

- In spite of the proper model for $p(\boldsymbol{a}_\tau|\boldsymbol{a}_{\tau-1})$, it does not prevent playing two frames of one library sound simultaneously. This can be forbidden by introducing the multinomial distribution (see in Appendix), however, the EKF cannot be used then.

3.7 Summary and Outline of Testing

In this chapter we presented a scenario for the complete automatic music transcription following the idea of an inverse music sequencer. It allows working with a bank of drum and harmonic music sounds as a memory base (the "sound library for estimation"). The music sounds in the observed audio signal are not identified only with the whole library sounds but also with their subparts. A probabilistic model containing unobserved variables of noise, amplitudes, labels (presences of sound segments – frames) and true inner library sounds was designed.

The variational Bayes technique was utilized to reduce the model equations and provide the algorithms for estimation of unobserved variables. It was shown (see in Section 3.5) that in the estimation of labels in our model

the variational Bayes outperforms the estimation (see the Extended Kalman Filtering algorithm in Section 3.6) in discrimination of the labels.

The estimation of noise, amplitudes and labels is a part of Algorithm 3 following from the base model and it is tested in Sections 4.8, 4.9. The base algorithm was extended by (i) Algorithm 5 for estimation of frames of the sound library and it is tested in Section 4.1; (ii) Algorithm 6 represents an online extension of the base algorithm with a benefit of distinct sound amplitude estimation in each time. In (ii), the algorithm testing is not a part of our experiments and it is kept as an option for a future work.

Other outputs of the thesis are worked on in the experimental part, in particular, (a) selection of evaluation measures, Algorithm 7 for our hit measure calculation – Section 4.3; (b) calculation of the nuisance parameters of the transition matrix $\boldsymbol{T}$ – Section 4.4; (c) comparison to multiple fundamental frequency estimation state-of-the-art – Section 4.10.

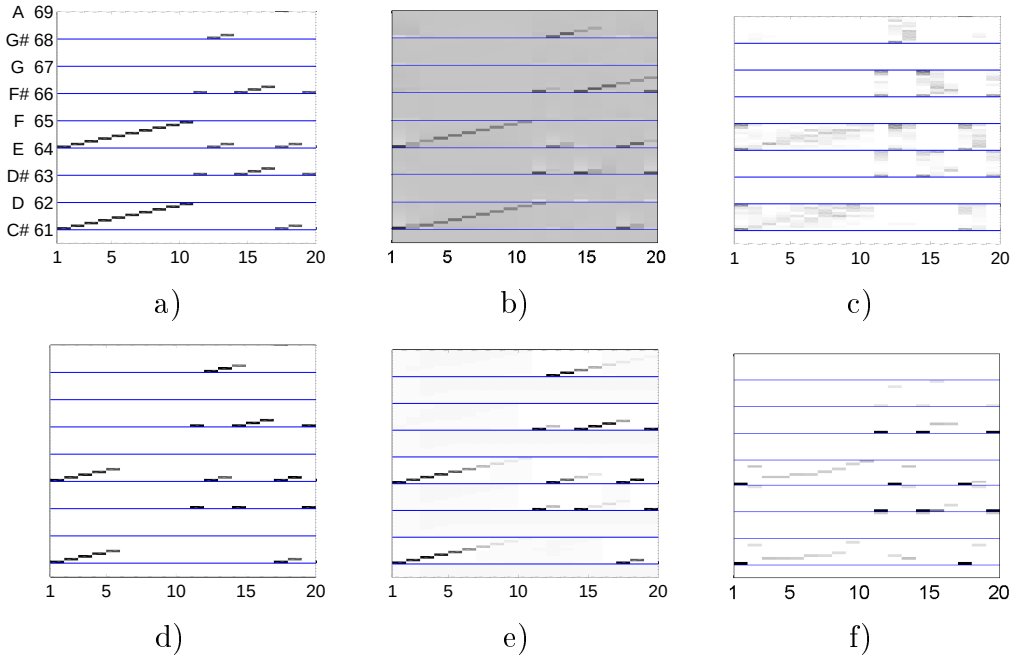


Figure 3.3: Example of simulated and transcribed piece of polyphonic music in the representation defined in Fig. 3.2. Vertical axes denote tone with the due midi keys. Horizontal axis denote discrete time (time units). Blue horizontal lines correspond to the beginning of the frame sequence due to a tone. **a)** original music excerpt; **b)** transcription via the extended Kalman filter model [68]; **c)** transcription via the NMF without any constraints (1.9); **d)** maximum values of the posterior estimates $\hat{\boldsymbol{L}}$ using the current model – result of our estimation using values of $\boldsymbol{T}$ having a higher value of the ratio $t_{end} : t_{next}$ than in the best case; **e)** posterior estimates of the current model $\hat{\boldsymbol{L}}$ without silence – the same not proper value of the ratio as in d); **f)** maximum likelihood $\hat{\boldsymbol{L}}$ estimates (without the transition part) using the current model.

Chapter 4

Experiments

Before we start to describe the elements of evaluation, let us denote the sound library, which the input observed signal was combined from, the "o-bank" or the ground truth sound library – $\boldsymbol{F}^{obs}$. The sound library, which the estimated signal was combined from, the "e-bank", is denoted by $\boldsymbol{F}^{est}$ as it is above.

4.1 Estimation of Sound Library Matrix

The possible benefit of the library estimation lies, e.g., in more robust estimation of $\boldsymbol{L}$ when the e-bank $\boldsymbol{F}^{est}$ is not selected properly. In this case the desired information is gathered from the observed data $\boldsymbol{y}_{1:t}$ of sufficient length t. The test of $\boldsymbol{F} = \boldsymbol{F}^{obs}$ estimation was performed on generated observed data $\boldsymbol{y}_{1:t}$. The purpose of the test was to estimate the length of the observed data $\boldsymbol{y}_{1:t}$ in order the sound library matrix $\boldsymbol{F}^{obs}$, label matrix $\boldsymbol{L}$ and amplitudes $\boldsymbol{a}$ to be estimated reasonably. The setting of the test was as follows: the balance coefficient (reflecting precision) $\xi \to 0$, number of tones $S = 3$ within the sound library, number of frames $k_s = 2$ within each sound s and number of spectral bins in $\boldsymbol{y}_\tau$, $\boldsymbol{f}_{s,i}$ was equal to 50. In order to suppress the affect of uncertainty in the observation model, we set $\omega \to \infty$ (reflecting precision). Necessary equations for the VB algorithm are presented in Section 3.3.

We assigned t to be long enough: either $= 160$ or $= 80$. All combinations of ones in the label vectors $\boldsymbol{l}_{s,\tau}$ can be repeated at least 2 times for $t = 80$. Both $t = 160$ and $t = 80$ yielded similar results: being $\boldsymbol{L}$ and $\boldsymbol{a}$ fixed, the sound library matrix $\boldsymbol{F}^{obs}$ could be reasonably estimated. As soon as $\boldsymbol{L}$ was estimated too and $\boldsymbol{a}$ was kept fixed, it resulted in these observations:

- Having $t \geq 80$ and the number of frequency bins greater than 50 did

not bring any improvement.

- Having $\xi = 0.0001$ (i.e., $\xi \to 0$) the order of frames $\boldsymbol{f}_{s,i}$ within $\boldsymbol{F}$ is not determined uniquely by the model. The value of ξ must not approach 0 and the e-bank $\boldsymbol{F}^{est}$ needs to be selected suitably in order the appropriate frame order within $\boldsymbol{F}$ to be estimated.

- The tests have proven that the model for $\boldsymbol{F}$ is not suitable for handling differences in magnitudes of peaks within the harmonic spectra between $\boldsymbol{F}^{est}$ and $\boldsymbol{F}^{obs}$. In Fig. 4.1, several tests for various ξ are depicted. It is shown that even for greater ξ the space of unknowns in $\boldsymbol{F}$ estimation is still large and if ξ is chosen even greater then $\boldsymbol{F}$ converges to $\boldsymbol{F}^{est}$.

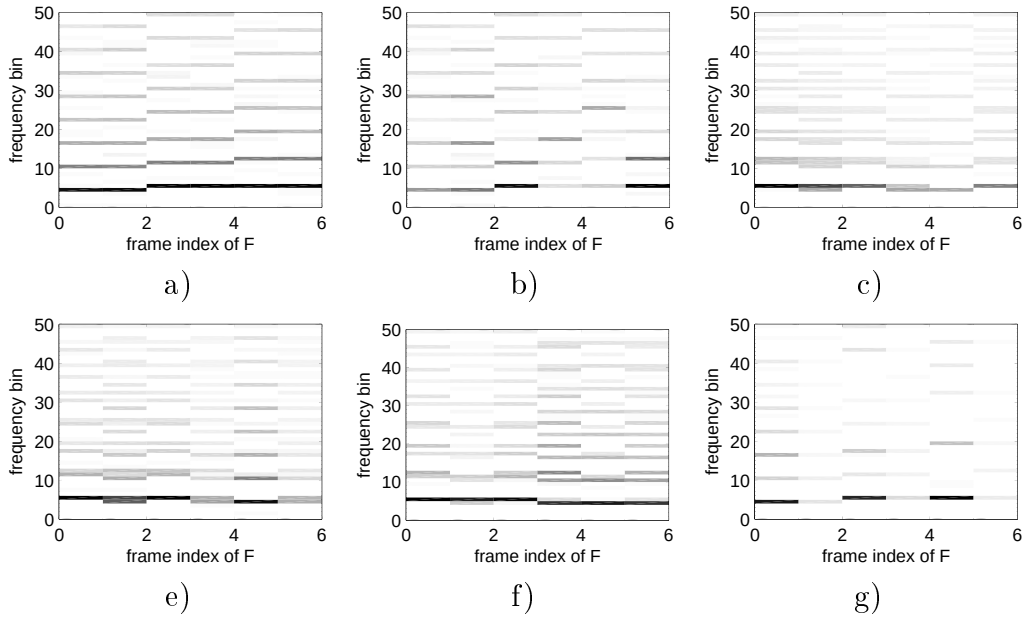


Figure 4.1: Estimation of the sound library $\boldsymbol{F}$ test. The ground truth library $\boldsymbol{F}^{obs}$ was generated to simulate a harmonic sound magnitude spectrum (a), using Matlab functions the e-bank was generated by $\boldsymbol{F}^{est} = rand(size(\boldsymbol{F}^{obs}))\cdot*\boldsymbol{F}^{obs}$ (b), the assessed libraries $\boldsymbol{F}$ for $\xi = 0.0001$, $\xi = 0.01$, $\xi = 1$, $\xi = 1000$ are depicted in (c) – (g), respectively.

In order the space in unknowns to be reduced the following modifications are proposed:

1. The variance for each frequency bin reflects the scaling matrix $\boldsymbol{D}_{\boldsymbol{F}_s,\phi}$

of $\boldsymbol{F}_s$:

$$p(\boldsymbol{F}_s) \propto \exp\left(tr\left\{-\frac{1}{2}(\boldsymbol{F}_s - \boldsymbol{F}_s^{est})^T \xi \boldsymbol{D}_{\boldsymbol{F}_s,\phi}^{-1}(\boldsymbol{F}_s - \boldsymbol{F}_s^{est})\right\}\right).$$

2. Two adjacent frames $\boldsymbol{f}_{s,i}$ and $\boldsymbol{f}_{s,i+1}$ within a component s should vary approximately as much as the varying between $\boldsymbol{f}_{s,i}^{est}$ and $\boldsymbol{f}_{s,i+1}^{est}$. This knowledge can be captured by creation of a new distribution, where the multiplication of $\boldsymbol{F}_s$ or $\boldsymbol{F}_s^{est}$ by matrix $\boldsymbol{U}$ (size $k_s x k_s - 1$) denotes shift of its columns to right and removing of the last column. Multiplication by matrix $\boldsymbol{V}$ (size $k_s \times k_s - 1$) denotes removing of the last column only:

$$\boldsymbol{U} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{V} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Imposing such phenomenon into the tested model yields:

$$p(\boldsymbol{F}_s) \propto \exp\left(tr\left\{-\frac{1}{2}\boldsymbol{\Sigma}_{\boldsymbol{F}_s}^{-1}(\boldsymbol{F}_s - \boldsymbol{F}_s^{est})^T(\boldsymbol{F}_s - \boldsymbol{F}_s^{est}))\right\}\right), \qquad (4.1)$$

where $\boldsymbol{\Sigma}_{\boldsymbol{F}_s} = (\xi_1 \boldsymbol{I} + \left[\xi_2(\boldsymbol{U} - \boldsymbol{V})(\boldsymbol{U} - \boldsymbol{V})^T\right])^{-1}$. The scalar ξ_2 denotes a new variable of a balance.

3. When the library sounds are represented by harmonic tones, frames of one tone do not differ significantly. Thus each frame could be represented as a linear combination of a small number of base vectors. The base vectors can be calculated by principal component analysis (see in Section 1.3) on $\boldsymbol{F}^{est}$. A necessary number of base vectors for a library sound representation can be assessed according to the number of significant eigenvalues.

The capturing of the proposed modification in our model would demand more tests. We decided this to be a subject of future work. In experiments of this thesis, various e-banks are chosen to test, i.e., the prior density $p(\boldsymbol{F}_s)$ is used from (3.40).

4.2 Simulated Data Testing Settings and Scheme

The simulated data were generated from piano midi files. Each note was represented by pitch, onset time, duration and offset in the sound library.

The offset is our extension of the midi format and it forms the truncation parameters. Such note parameters were represented by the ground truth label matrix $\boldsymbol{L}^{gt}$ and by the ground truth vector of amplitudes $\boldsymbol{a}^{gt}$. Midi notes, which were not available in the library of sounds, were omitted and the notes longer than[1] $k_s = 10$ were truncated to this length. Since one piano cannot play the same note simultaneously, it is ensured that there will be at most one active frame of the same sound at time t.

The o-bank was made from 61 library sounds (corresponding to midi notes 36 – 96), each of them having $k_s = 10$ window. Each window contained 4096 samples at 44.1 kHz sample rate and the windows did not overlap. After the STFT and taking the absolute values from the complex spectra we obtained 2048 relevant samples. Considering the first half of frequency bins follows from the fact that all bins higher than the Nyquist frequency are affected by *aliasing* phenomenon [90]. The first 600 frequency bins of these were taken to represent the frame (a magnitude spectrum) $\boldsymbol{f}_{s,i}$. This approximately corresponds to 6 kHz for the highest frequency bins which was used in tests of multiple fundamental frequency detector of Klapuri [16, 51]. The same processing was applied on the observed signal $time(\boldsymbol{y}_\tau)$, i.e., the time domain signal before processing by the STFT.

Remark. Contrary to Klapuri we did not consider the processes of the human ear and human brain model, except of the compression which is represented by scaling in our model.

Following the scheme in Fig. 4.2, the observed audio signal was generated using model (3.1). Since the observed audio signal is generated according to $\boldsymbol{L}^{gt}$, $\boldsymbol{a}^{gt}$, it does not contain any subsiding sound or a sound of a tone onset outside the frame sequences. The observed audio signal is fed into Algorithm 3 along with the nuisance parameters $\boldsymbol{\delta} = [r_\omega, \sigma_{\mu,0}, \sigma_{a,0}, t_{sil}, t_{end}, t_{start}, t_{next}]$ and the e-bank. The e-bank contained the same number of sounds, however each of them can have more than $k_s = 10$ non-overlapping frames – in our experiments up to 50. The label matrix $\hat{\boldsymbol{L}}$ and amplitudes $\hat{\boldsymbol{a}}$ are estimated by Algorithm 3. In evaluations, the quantities $\boldsymbol{L}$,$\hat{\boldsymbol{L}}$, $\hat{\boldsymbol{a}}$,$\hat{\boldsymbol{a}}^{corr}$ are used. The difference between $\boldsymbol{L}$ and $\hat{\boldsymbol{L}}$ lies in that that the label $\boldsymbol{L}$ represents the value of the phenomenon and contains zeros and ones only whereas $\hat{\boldsymbol{L}}$ represents the mean of the phenomenon and can contain any real numbers, see the multinomial distribution definition in Appendix and Section 4.3 to describe the value assignment to $\boldsymbol{L}$. The introduction of a corrected estimate of amplitudes $\hat{\boldsymbol{a}}^{corr}$:

[1] If the sample rate is 44.1 kHz, the STFT window length is 4096 and the windows are not overlapping, then $k_s = 10$ corresponds to one second of a recording approximately.

$$\hat{\boldsymbol{a}}^{corr} = c_{\boldsymbol{a}} \cdot \hat{\boldsymbol{a}} \tag{4.2}$$

is a consequence that the superposition principle does not hold for magnitude spectra, see in Subsection 3.1.1. There are two evaluation approaches: (i) a measure calculated from the estimated quantities (hit measures, amplitudes) and (ii) a measure calculated from the audio signals which were made on the basis of the quantities (sound-to-distortion ratio – SDR).

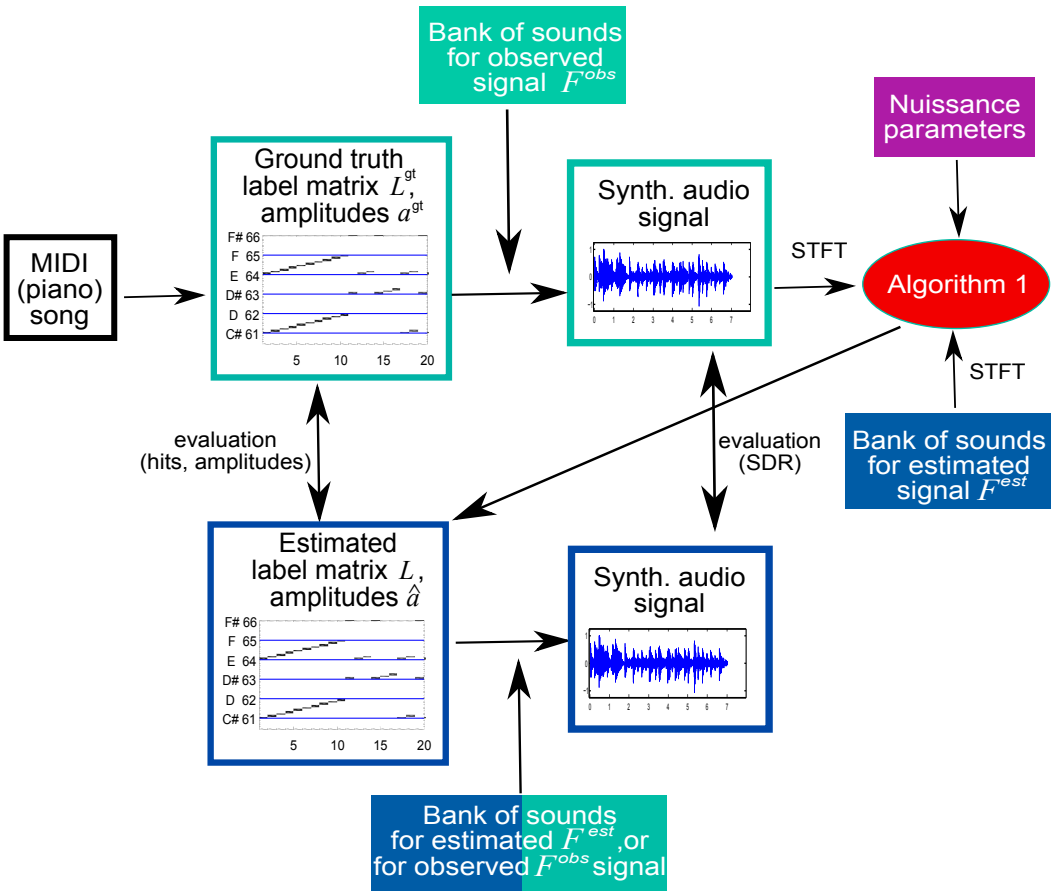


Figure 4.2: Evaluation scheme

4.3 Evaluation Measures

The goal of Algorithm 3 is to maximize the number of hits of frames, the SDR measures and minimize the error measure in amplitudes.

Hit Measure

We expect that there is at most one active frame at each time τ within the estimated signal, therefore the estimated label matrix of zeros and ones, $\boldsymbol{L}$, is used in hit measure calculations (not the matrix $\hat{\boldsymbol{L}}$). The hit measure represents the number of estimated tone frames (from the e-bank) corresponding to the ground truth tone frames (from the o-bank). It is characterized by "hits" m_h as a number of correctly hit labels, "false-positives" m_{fp} as a number of estimated labels that are not correct and "false-negatives" m_{fn} as the missing correct labels that should have been estimated. Let us refer to these quantities as *measures #1.*

The phases of the complex STFT spectrum of $time(\boldsymbol{y}_t)$ vary a lot over time [1]. In the sounds of natural musical instruments they even vary a lot from one recorded sample to another. Therefore, in our experiments, it can happen very easily, e.g., that for i-th frame $\boldsymbol{f}_{s,i}$ of the tone A_2 from the e-bank and i-th frame $\tilde{\boldsymbol{f}}_{s,i}$ of the corresponding tone A_2 from the o-bank (ground truth), the observational probability is higher for some other j-th frame $\boldsymbol{f}_{s,j}$, $i \neq j$, of the same A_2 than for the i-th. In other words, in a real world often there exists a frame index within a sound from the e-bank that resembles more to some other index than to that ground-truth one. It follows that the estimated frame sequences within a sound often hit a correct sound, have a correct length but do not start by the correct frame index. Sounds in polyphony come even more under such behavior.

Hence, this situation should not be marked as an error. Therefore we propose the hit *measure #2.* Let us denote the correctly hit labels m_{h2} and the missing labels m_{fn}. The false-positives are split into two measures – the "false-positives-within-the-sound" m_{fp2s} and the "false-positive outliers" m_{fp2o}. The measures #2 are calculated by Algorithm 7. In Fig. 4.3 there is a label matrix $\boldsymbol{L}$ excerpt . The yellow squares denote the active frames within the estimated label matrix $\boldsymbol{L}$, whereas the black squares represent the active frames within the ground truth matrix $\boldsymbol{L}^{gt}$. At discrete times 3 and 8, we recorded the false-positive-within-sound active frames, they were gathered into m_{fp2s} and at $10, 11$ the false-positive outliers were identified and summed into m_{fp2o}. The rest of yellow squares correspond to hits $m_{h2} = 6$. There is no false-negative gathered into m_{fn2} and for the distance $M_{d_{s,\tau_0}}$ we have $M_{d_{s,\tau_0}} = 4$. When the hit measures #1 were calculated there would not be any hit, i.e., $m_h = 0$, but just all false-positives represented by $m_{fp} = 8$.

The false-negatives of hit measure #2 m_{fn2} and the number of false-positive outliers m_{fp2o} represent suitable quantities of error measure of the label identification for a real case. They are used to calculate the overall hit error rate in Section 4.6. The false-positives-within-the-sound m_{fp2s}

Algorithm 7 Hit measure #2 calculation.

1. Initialize $m_{h2} = 0$, $m_{fn2} = 0$, $m_{fp2s} = 0$, $m_{fp2o} = 0$.

2. Find all active frame sequences in the ground truth label matrix $\boldsymbol{L}^{gt}$. The active frame sequence is a representative of a tone playing without an interruption. If there exists an active $l^{gt}_{s,i,\tau+1}$ for any i from the same sound s (without the silence frame), it belongs to the frame sequence ending up by an active $l^{gt}_{s,j,\tau+1}$ where j denotes a sound frame of the sound s .

3. For all active frame sequences in all sounds s

   (a) For all times $\tau = \tau_0, \ldots, \tau_{end}$ of the sequence

      i. Calculate the vector elements $d_{s,\tau_0-\tau+1}$ of distances between active frames of the estimated sequence and active frames of the ground truth sequence, that is, $d_{s,\tau_0-\tau+1} = h(l_{s,\tau}) - h(l^{gt}_{s,\tau})$. Here h applied either on the estimated or the ground truth labels denotes the function that returns the active frame index within the sound s.

   (b) Get the most occurring distance $M_{d_{s,\tau_0}}$ among $d_{s,\tau_0-\tau+1}, \forall\tau$.

   (c) For all times $\tau = \tau_0, \ldots, \tau_{end}$ of the sequence

      i. if $M_{d_{s,\tau_0}} + h(l^{gt}_{s,\tau}) = h(l_{s,\tau})$ then $m_{h2} = m_{h2} + 1$,
      else $m_{fp2s} = m_{fp2s} + 1$.

4. $m_{fp2o} =$ "all active frames in $\boldsymbol{L}$" $- m_{h2} - m_{fp2s}$,
   $m_{fn2} =$ "all active frames in $\boldsymbol{L}^{gt}$" $- m_{h2}$.

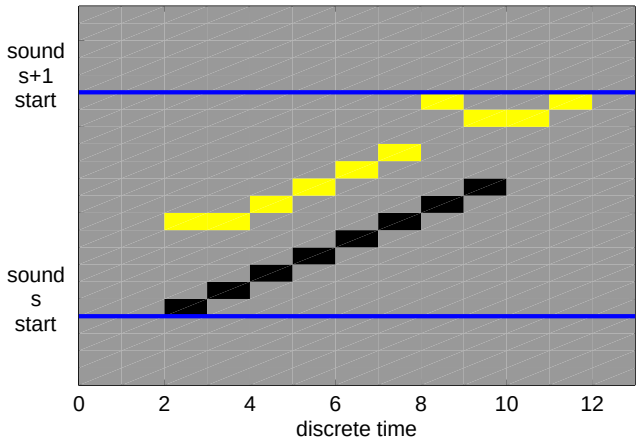


Figure 4.3: Hit measure #2 calculation – example

quantity's risk is not so serious when considering the fundamental frequency estimation problem. It would become more serious if frames within a sound did not resemble that much as they are when they correspond to harmonic sounds.

SDR

In order to provide a similarity measure of the resulting audio signal to the ground truth audio signal we calculated the total relative sound-to-distortion ratio measure [61, 91].

We utilized three approaches in the SDR calculation (termed 1., 2., 3., respectively). Each consists of the time and the frequency option (termed (a), (b), respectively). The calculation formula common for all of them reads

$$\boldsymbol{SDR} = 10 \cdot \log_{10} \frac{\sum_{j\tau}[\hat{s}_{j\tau}]^2}{\sum_{j\tau}[b \cdot s_{j\tau} - \hat{s}_{j\tau}]^2}, \tag{4.3}$$

where b is a scalar fitting[2] $\boldsymbol{s}_\tau = b \cdot \hat{\boldsymbol{s}}_\tau + noise$ where $noise \sim \mathcal{N}(0, c \cdot \boldsymbol{I})$. The calculated constant b can produce as good or better estimates of SDR as $\frac{1}{c_{\boldsymbol{a}}}$, therefore $\hat{\boldsymbol{a}}^{corr}$ is not considered in calculations here and $\hat{\boldsymbol{a}}$ is used instead.

1. *MAP estimate #1.* The position of the one in the label vector $\boldsymbol{l}_{s,\tau} = [0, 0, \ldots, 1, \ldots, 0]^T$ is determined by the maximum value from the elements of $\hat{\boldsymbol{l}}_{s,\tau}$. We note that by the *time* operator in the following equations it is meant that the original time domain signal is used instead of the magnitude STFT spectrum. The operator $mstft$ implies

[2] The calculation of b is accomplished by the least squares method.

the transforming of the time domain signal into the spectral domain by the STFT and using its magnitude spectrum.

   (a) $\hat{\boldsymbol{s}}_\tau = time(\boldsymbol{F}^{est})\hat{\boldsymbol{A}}\boldsymbol{L}_\tau$, $\boldsymbol{s}_\tau = time(\boldsymbol{y}_\tau) = time(\boldsymbol{F}^{obs})\hat{\boldsymbol{A}}\boldsymbol{L}^{gt}_\tau$.
   (b) $\hat{\boldsymbol{s}}_\tau = \boldsymbol{F}^{est}\hat{\boldsymbol{A}}\boldsymbol{L}_\tau$, $\boldsymbol{s}_\tau = \boldsymbol{y}_\tau = mstft(time(\boldsymbol{F}^{obs})\hat{\boldsymbol{A}}\boldsymbol{L}^{gt}_\tau)$.

2. *MAP estimate #2.* Contrary to MAP estimate #1, the o-bank $\boldsymbol{F}^{obs}$ is used for the estimated time domain signal calculation $\hat{\boldsymbol{s}}_\tau$.

   (a) $\hat{\boldsymbol{s}}_\tau = time(\boldsymbol{F}^{obs})\hat{\boldsymbol{A}}\boldsymbol{L}_\tau$, $\boldsymbol{s}_\tau = time(\boldsymbol{y}_\tau)$.
   (b) $\hat{\boldsymbol{s}}_\tau = \boldsymbol{F}^{obs}\hat{\boldsymbol{A}}\boldsymbol{L}_\tau$, $\boldsymbol{s}_\tau = \boldsymbol{y}_\tau$.

3. *Mean estimate.* The label matrix mean values $\hat{\boldsymbol{L}}$ are used along with the e-bank $\boldsymbol{F}^{est}$.

   (a) $\hat{\boldsymbol{s}}_\tau = time(\boldsymbol{F}^{est})\hat{\boldsymbol{A}}\hat{\boldsymbol{L}}_\tau$, $\boldsymbol{s}_\tau = time(\boldsymbol{y}_\tau)$.
   (b) $\hat{\boldsymbol{s}}_\tau = \boldsymbol{F}^{est}\hat{\boldsymbol{A}}\hat{\boldsymbol{L}}_\tau$, $\boldsymbol{s}_\tau = \boldsymbol{y}_\tau$.

Amplitudes

Two cases are dealt: (i) there is one common amplitude for all components which is not changed over time, (ii) each component can have its own amplitude which is not changed over time. In the former case, the amplitude is estimated by running Algorithm 3 with the settings defined in Subsection 4.9.3, while in the latter case, with the settings defined in Subsection 4.9.2. Because in the latter case the amplitudes to be estimated have the same value, we used their standard deviation as an assessment of the estimation.

Summary on SDR, Hit Measures and Amplitudes

Evaluation methods reflecting measures of hits in labels and sound-to-distortion values were proposed. Both measures are important – the "hits" represent accuracy in the target music content information retrieval while the SDR assesses the overall audible result with respect to prior information largely influenced by the sound library selection. When we take a library sound as a whole, it always corresponds to the whole ground truth library sound, most of all library sounds (see explanation in Section 4.5). However the resemblance does not hold for the frames within the sound, therefore we proposed Algorithm 7 for a new hit measure calculation. The SDR and both types of hit measures are applied in the evaluation of the proposed algorithms in Section 4.8 and 4.9. For the amplitudes, we calculate their common amplitude; or, estimate the amplitudes individually and provide their standard deviation as an estimation evaluation.

4.4 Nuisance Parameters of the Transition Matrix

The transition values $[t_{sil}, t_{end}, t_{start}, t_{next}]$ can be obtained by maximizing the posterior distribution of $\boldsymbol{L}^{gt}$ in the training phase. Given $\boldsymbol{L}^{gt}$, the observed data $\boldsymbol{Y}$ and the transition matrix $\boldsymbol{T}$ are mutually conditionally independent, therefore the transition values can be calculated by seeking a maximum of the probability density on $\boldsymbol{T}$ only. It corresponds to maximizing the function

$$g(\boldsymbol{T}|\boldsymbol{L}^{gt}) = \sum_{\tau,s} \boldsymbol{l}^{gt}_{s,\tau} \log \boldsymbol{T}\, \boldsymbol{l}^{gt}_{s,\tau-1}.$$

Moreover, the parameters can be obtained directly by summing up transitions in $\boldsymbol{L}^{gt}$ in order to avoid using optimization algorithm like Matlab's fminsearch. There we have two possibilities: either to sum up transitions $[t_{sil}, t_{end}, t_{start}, t_{next}]$ individually for each tone, or to sum up the transitions over all tones. In our experiments, we utilized the former approach since in most cases, it provided around 3% improvement in the hit #2 measures.

4.5 Simulation Data and Sound Libraries

This section contains a description of the simulation data and sound libraries, we do not bring any contribution of the thesis in this section, we only name main properties of the input data. Due to a computation load, the length of time for estimation tests was chosen up to six minutes, see in Section 4.7.

The observed music signal was made by using the o-bank. The feasible library of sounds for estimation, the e-bank, needs to satisfy these conditions: when we take a library sound as a whole, it always corresponds to the whole sound of the ground truth library (o-bank), most of all its library sounds. We define that the correspondence is determined by the approximation of the Poisson distribution, i.e., $\forall s, \sigma \in S, \sigma \neq s: \quad \mathcal{P}o(\boldsymbol{F}^{est}_{\sigma}|\boldsymbol{F}^{obs}_{s}) < \mathcal{P}o(\boldsymbol{F}^{est}_{s}|\boldsymbol{F}^{obs}_{s})$. The correspondence is not required for the frames within the sound. We have to note that neither one of the frames $\boldsymbol{f}^{obs}_{s,i}$, $\boldsymbol{f}^{est}_{s,i}$ starting by $i = 2$ is a zero vector. The $i = 1$ is reserved for the frame of silence, see in Subsection 3.1.1.

Sound Libraries

In experiments we used the following sound libraries of a piano:

1. Sound library #1 (SL #1) – University of Iowa Piano – mezzo-forte [92] (Fig. 4.4),

2. Sound library #2 (SL #2) – University of Iowa Piano – forte [92] (Fig. 4.4),

3. Sound library #3 (SL #3) – 4Front Piano Module Free VST library [93] (Fig. 4.5),

4. Sound library #4 (SL #4) – 4Front E-Piano Module Free VST library [93] (Fig. 4.5).

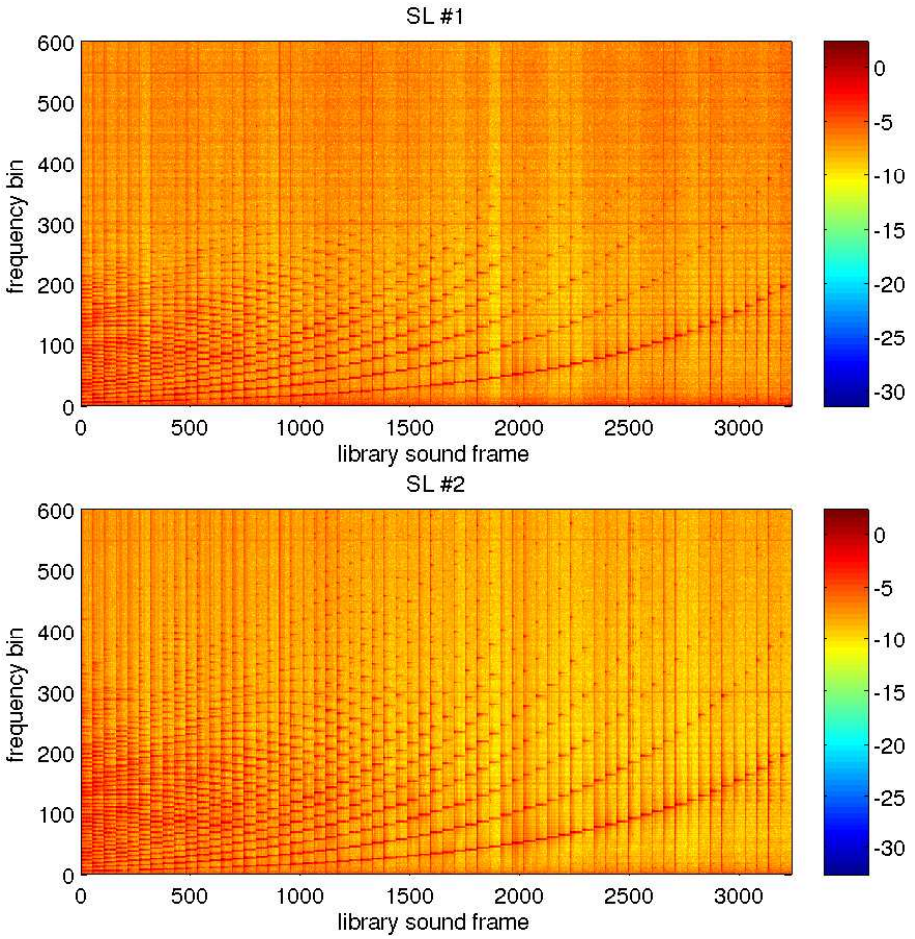


Figure 4.4: Sound library #1 and #2 represented by $\ln(\boldsymbol{F})$. In $\boldsymbol{F}$, a few values are around e^{-30}.

Simulation Data

In Fig. 4.7, simulation data characteristics are depicted. In the picture a) there are $[t_{sil}, t_{end}, t^*_{start}, t_{next}]$ for all components. Since the onset of a note

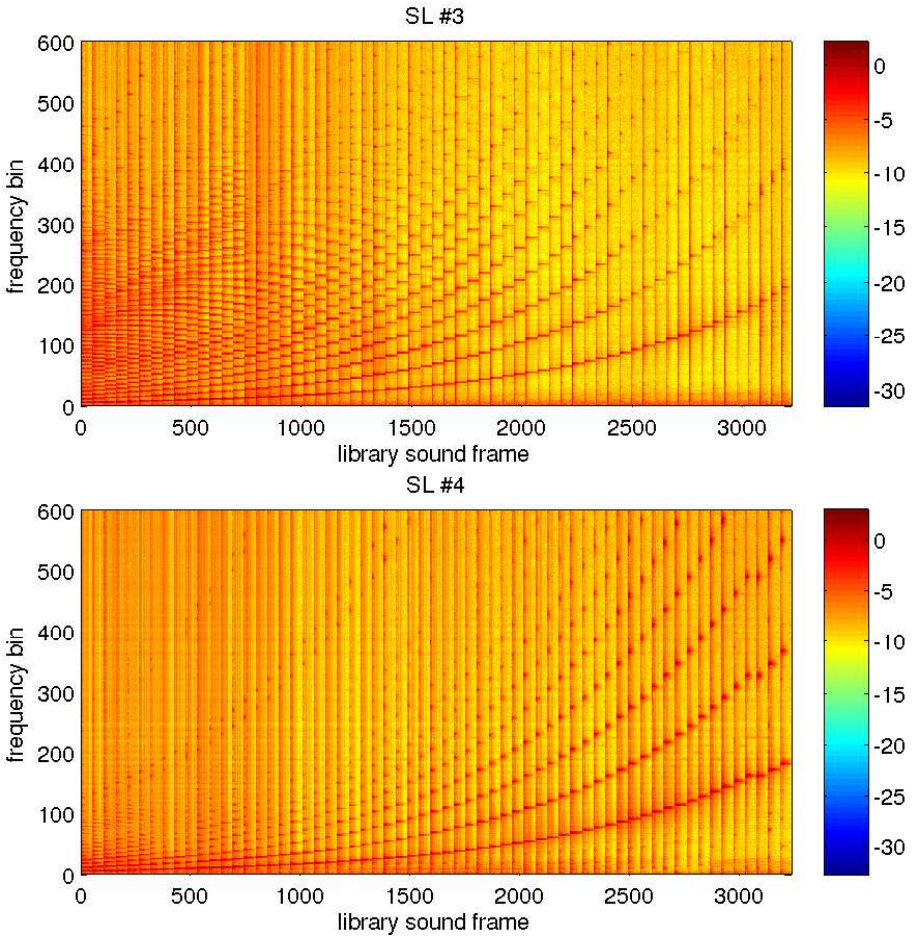


Figure 4.5: Sound library #3 and #4 represented by $\ln(\boldsymbol{F})$. A few values in $\boldsymbol{F}$ are around e^{-30}.

in a MIDI file does not contain information where to start in the library sound, we chose, in the label matrix $\boldsymbol{L}^{gt}$, for all note onsets to start by the first sound frame. Therefore, when we calculate t_{start} in all elements of the transition matrix $\boldsymbol{T}$ using such $\boldsymbol{L}^{gt}$ we get only the first top t_{start} element of non-zero value. Since the note onset frame can be anywhere within a sound s, we decided the obtained non-zero probability t_{start} to be termed t^*_{start} and it is independently identically distributed among all t_{start} of $\boldsymbol{T}$, thus $t_{start} = t^*_{start}/k_s$.

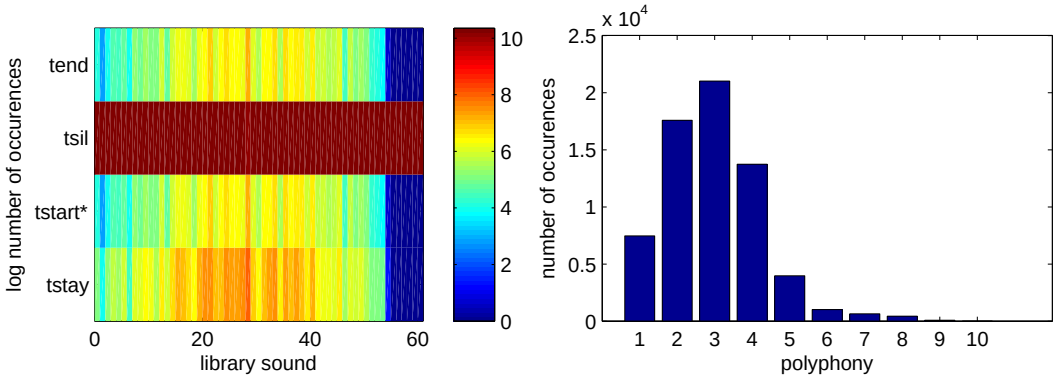
In the picture b) of Fig. 4.7 there is the polyphony characteristic and in the picture c), d), e), f) there are values of $diag(\boldsymbol{D}_\phi)$ used to scale the frequency bins of $\boldsymbol{y}_t$.

Simulation Data #1 are the longest and in our experiments, we only use

$[t_{sil}, t_{end}, t_{start}, t_{next}]$ obtained from this. In the figures, we can notice that t^*_{start} and t_{end} are identical and neither one of the simulation data contains any sounds after 56th tone of the 61. In order to enable the library sounds 57 – 61 estimation in our experiments, we assigned ones to the four-duple $[t_{sil}, t_{end}, t^*_{start}, t_{next}]$ of the sounds 57 – 61. In the pictures of polyphony, the "occurrence" concerns a library sound frame, not the whole polyphony. The number of a distinct polyphony can be obtained simply by division of a number-of-occurrence bar by the polyphony number. In the pictures of $diag(\boldsymbol{D}_\phi)$, we can see oscillations. Calculating $diag(\boldsymbol{D}_\phi)$ on longer observed data and using a variety of musical instruments in the library results in the oscillations occurring at 12 cycles per octave [43].

The list of simulation data characteristics is as follows:

- Simulation Data #1 (SD #1) – Fig. 4.6; length 48 minutes, 35 seconds.
  - Mozart: * Sonata No. 11 A major (Alla Turca) , KV 331 (1783) (13:52, 6:06, 3:11)
  - Debussy:
    - I. Doctor Gradus ad Parnassum Modérément animé 2:24
    - Jimbo's Lullaby Assez modéré 3:09
  - Bethoveen:
    - 1. Movement Allegro molto e con brio 6:31
    - 2. Movement Adagio molto 6:32
    - 3. Movement Prestissimo 3:37
  - Bach: * Prelude and Fugue in C minor BWV 847



a) b)

Figure 4.6: Simulation Data #1 characteristics

- Simulation Data #2 (SD #2) – Fig. 4.7.
  - Debussy: I. Doctor Gradus ad Parnassum Modérément animé 2:24

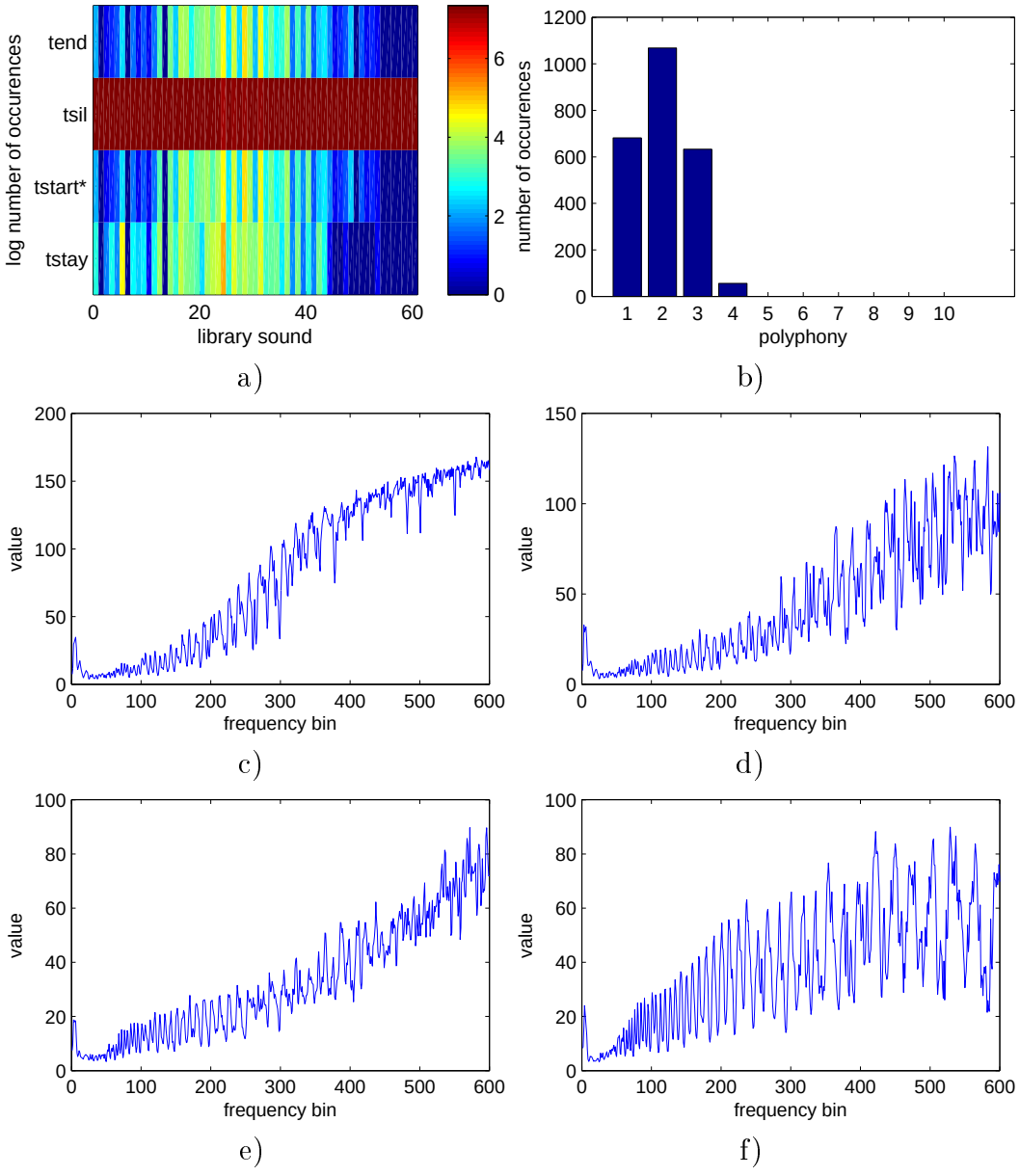


Figure 4.7: Simulation Data #2 characteristics. The pictures c), d), e), f) denote $diag(\boldsymbol{D}_\phi)$ gained from simulation data generated from SL #1, SL #2, SL #3, SL #4, respectively.

- Simulation Data #3 (SD #3) – Fig. 4.8; length 5 minutes, 18 seconds.
  - Debussy: Clair de Lune Andante tres express; length: 1:02
  - Chopin: No. 7 Andantino; length: 0:36
  - Mozart: 2. Movement Andante – part, length; 0:40
  - Bach: Prelude and Fugue in C major BWV 846; length: 1:46
  - Beethoven: For Elise Poco moto; length: 1:14

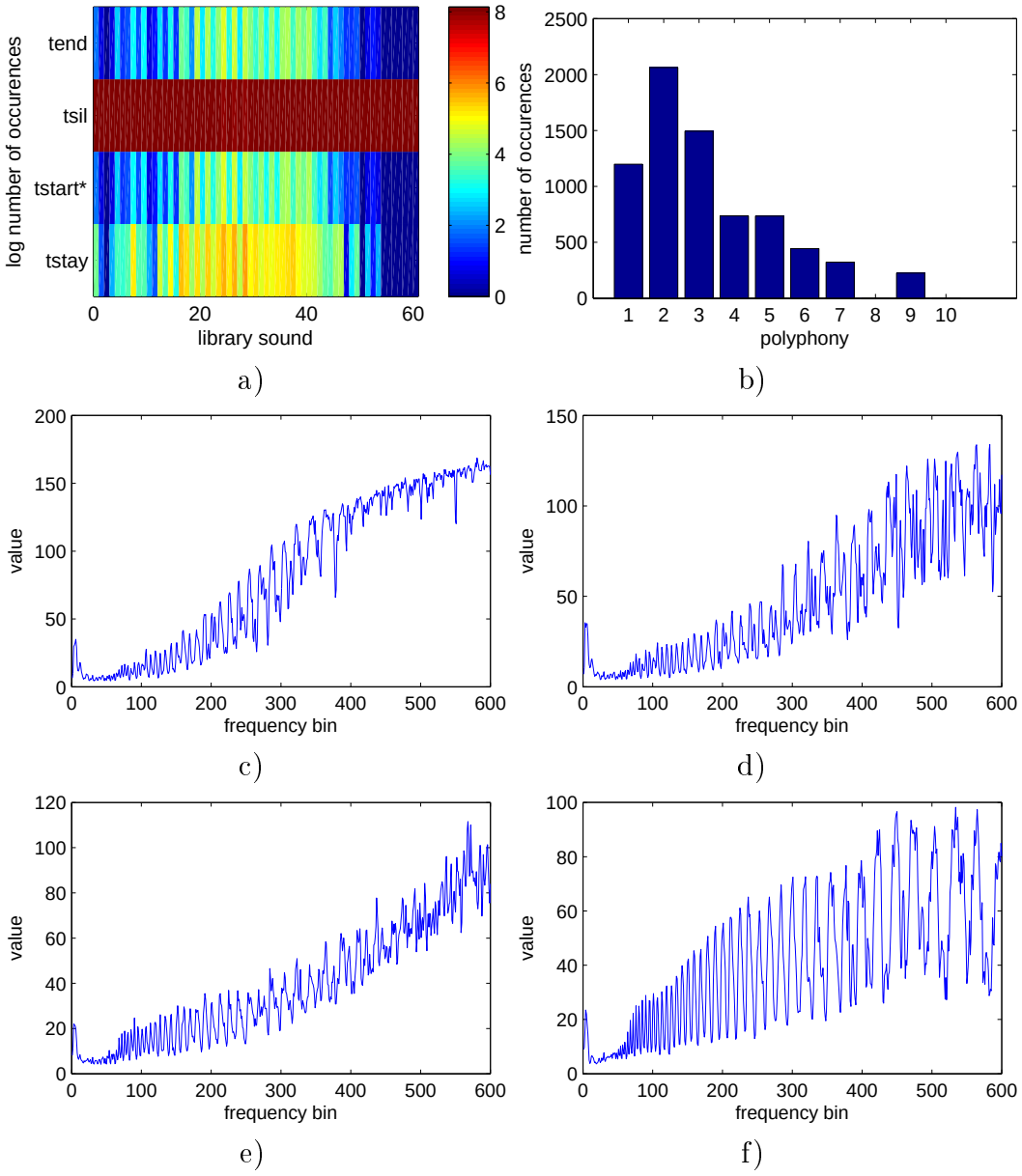


Figure 4.8: Simulation Data #3 characteristics. The pictures c), d), e), f) denote $diag(\boldsymbol{D}_\phi)$ gained from simulation data generated from SL #1, SL #2, SL #3, SL #4, respectively.

- Simulation Data #4 (SD #4) – Fig. 4.9; 50 tones on 500 frames (top 11 of 61 tones from the sound library are omitted).

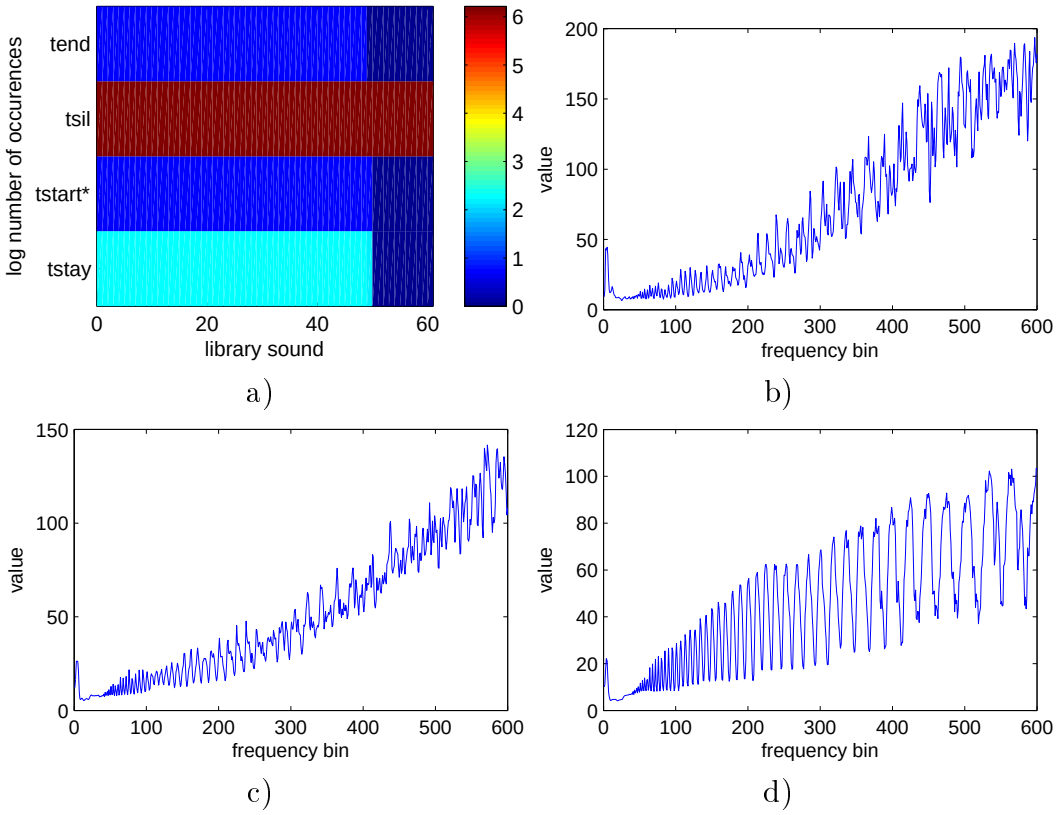


Figure 4.9: Simulation Data #4 characteristics. The pictures b), c), d) denote $diag(\boldsymbol{D}_\phi)$ gained from simulation data generated from SL #2, SL #3, SL #4, respectively.

4.6 Descriptions of Figures with Results

In Fig. 4.10 there are types of result representations which can be met in Subsections 4.8, 4.9.

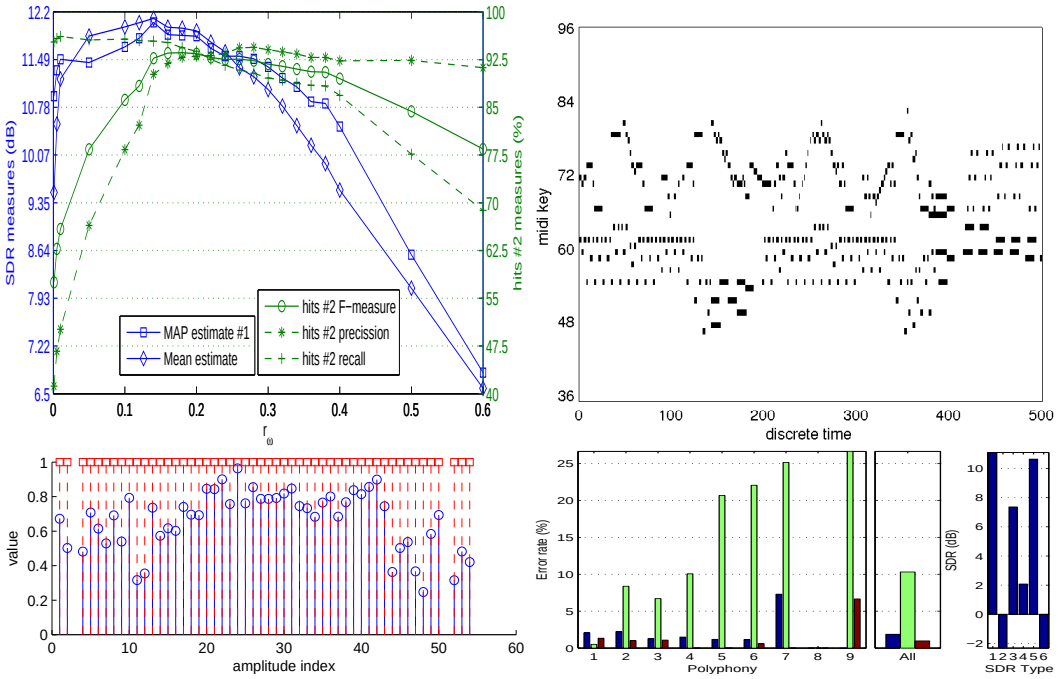


Figure 4.10: Types of representation of results

Top-left picture represents SDR measures and hits #2 measures as a function of the threshold r_ω. The hit #2 measures are transformed from $[m_{h2}, m_{fn}, m_{fp2s}, m_{fp2o}]$ to precision P_2, recall R_2 and F-measure F_2 defined as

$$P_2 = (m_{h2} + m_{p2s})/(m_{h2} + m_{p2s} + m_{fp2o}) \cdot 100\% \quad (4.4)$$

$$R_2 = (m_{h2} + m_{p2s})/(m_{h2} + m_{p2s} + m_{fn}) \cdot 100\% \quad (4.5)$$

$$F_2 = 2 \cdot (P_2 \cdot R_2)/(P_2 + R_2) \cdot 100\% \quad (4.6)$$

The precision measure can be simply understood as a normalized measure of false-positive outlier quantity m_{fp2o} and recall as a normalized measure of missing frames m_{fn}. In order to provide one overall measure in hits, the F-measure F_2 was defined. In the text there are also SDR measures and hits #2 as a function of $\sigma_{a,0}$ in Fig. 4.20 in Section 4.9.

Top-right is a representation of $\hat{\boldsymbol{L}}$ in a piano-roll instead of the representation without any transformation as it is in Fig. 3.2. The transformation of representation of $\hat{\boldsymbol{L}}$ in Fig. 3.2 into the piano-roll is accomplished by omitting information of frame index i within a sound s and by displaying only the maximum value from each vector $\hat{\boldsymbol{l}}_{s,\tau}$. The colors in the piano-roll image are selected according to Matlab "gray" color map where the white one

is assigned to zero value and the black color to a value one. The values – probabilities in $\hat{\boldsymbol{L}}$ are not a representation of amplitudes.

Bottom-left: red squares on value 1 denote that any frame from the sound index s is present in the ground truth. Hence, when there is no red square but a blue circle, we have some frames from the sounds not present in the ground truth; and, when the situation is vice versa, the sound is completely without presence in the testing simulation data. In Section 4.9, the standard deviation of the blue circled values, which matches the positions of the red squares, is calculated. If, in its calculation, a blue circle is missing on the position of a red square, the value corresponding to the missing circle is assigned to zero.

Bottom-right: The two left panels contain error-rate measurements of precision P_2 (the blue bar), recall R_2 (the green bar) and of the error measure of false-positives-within-the-sound in all frames in the correct sound $H_{2,err}$ (the red bar):

$$H_{2,err} = \left(1 - \frac{m_{fp2s}}{m_{h2} + m_{fp2s}}\right) \cdot 100\%. \tag{4.7}$$

The left panel with error-rate measurements contains P_2, R_2, H_2 for each polyphony and the right panel with error-rate measurements displays them overall. The right panel with SDR measures contains several values: 1 – MAP estimate #1 (a), 2 – MAP estimate #1 (b), 3 – MAP estimate #2 (a), 4 – MAP estimate #2 (b), 5 – Mean estimate (a), 6 – Mean estimate (b). They were defined in Section 4.3 – SDR.

4.7 Computational Load

One run of Algorithm 3 (10 VB-iteration cycles) on Simulation Data #2 of length 5 minutes, 15 seconds, using the library of sounds for estimation (e-bank) with sixty-one 5-second long sounds, took approximately 2 hours and 15 minutes on a cluster computer with a six-core AMD processor in Metacentrum[3]. The program was written in Matlab. The calculation of amplitudes took 5% and the calculation of labels 95% from the whole algorithm running time. The computational load increased linearly with: (i) increase of frames in the library, (ii) increase of the length of time of the observed signal.

The huge computational load is caused by large data: Simulation Data #2 with 5-second sounds in the library is represented by the label matrix $\boldsymbol{L}$

[3]Catch-all MetaCentrum Virtual Organization operates and manages distributed computing infrastructure consisting of computing and storage resources owned by CESNET as well as those of co-operative academic centers within the Czech Republic.

of more than 10 million labels which must be assessed in each VB-iteration. The computational savings can be achieved by excluding the sounds which are not present in the observation, e.g., in a pre-processing stage. Moreover, the observation signal can be split into shorter windows containing just a several tones, in sequel the pre-processing can save even more of the load. This is a part of a future work.

4.8 Tests without Estimation of Amplitudes

In order to estimate, without amplitudes, Algorithm 3 is used so that the amplitude mean value $\mu_{hyp,a,0}$ is being fixed to the assessed best estimate and $\sigma_{a,0}$ is selected close to zero. Since the best estimate value is not known (because the superposition principle does not hold here – see eq. (3.3)), the amplitude value $\mu_{hyp,a,0}$ can be estimated (i) as in Subsection 4.9.3 or (ii) by the estimation of the coefficient b from (4.3). In this subsection we selected $\mu_{hyp,a,0} = 0.65$ for all tests, which is, from (4.2) we have $c_{\boldsymbol{a}} = \frac{1}{0.65}$. The $max\_iterations$ in Algorithm 3 was set to 10.

Each of the following subsections represents the change in one aspect while the others are retained. The aspects are

1. observed signal,
2. sound library,
3. length of observed signal,
4. length of sounds,
5. scaling in frequency vs. no-scaling at all,
6. time and frequency scaling vs. frequency scaling only,
7. exact fit tests.

4.8.1 Change in the Observed Signal While Other Parameters Not Changed

In the following figures there are three triples of tests according to three o-bank and e-bank combinations. The tests inside each triple can be compared to each other since, except for the observed signal, all other parameters are the same. Fig. 4.11 contains the first triple and represents tests when o-bank is SL #1, e-bank is SL #2, i.e., the most fitting case of o-bank and e-bank;

the second triple in Fig. 4.12 is tested when o-bank – SL #3, e-bank – SL #1, i.e., the less fitting case of synthesized piano vs. true piano; the third triple in Fig. 4.13 when o-bank – SL #4, e-bank – SL #1, represents the case when the timbre of e-bank and the o-bank are much different, see in Fig. 4.4, 4.5.

The triple is a test of a sequence of one second long notes and two different parts from SD #3. The left column contains tests on SDR and hit #2 measures whereas the right column is a piano-roll representation of the test for $r_\omega = 0.24$ from the left column. The first row: the note sequence. There are 50 tones from 61 on 500 length, the rest of 11 tones is omitted; the second row: Mozart part from SD #3; the third row: Debussy part from SD #3, all of them of the same length of 500 frames. The two 500-frames-long parts from SD #3 were taken from the part of Mozart, Debussy. Regarding all hit and SDR measures contained in the graphs, we conclude that the Debussy piece is more difficult to transcribe than the one of Mozart. In spite of the fact that the sequence of notes is a monophony, it yields worse results both in the SDR and the hit #2 measures than the two pieces of Mozart and Debussy. It can be explained by a different transition distribution in the note sequence than the one characterized in $[t_{sil}, t_{end}, t_{start}, t_{next}]$ and by higher number of distinct notes in the observed signal.

4.8.2 Change in the Sound Library While Other Parameters Not Changed

This subsection was tested on SD #2. In Fig. 4.14, it can be seen the dependency of SDR and hit #2 measures on the used sound library, other settings for the 5-duple tests were not different. We conclude that the switch between o-bank and e-bank does not lead to a significant change in the measured values. One of the examples is using the shifted version of the o-bank in the e-bank. That is, total length of SL #1 sounds is more than 6 seconds. The o-bank is made up of the first second of the 6 seconds and the e-bank is made up of the remaining 5 seconds. From the results, we can see that using the same set of sounds – natural harmonic tones – for the creation of the o-bank and the e-bank does not necessarily lead to better results than if the o-bank and the e-bank were created from different sounds.

4.8.3 Change in Length of Library Sounds While Other Parameters Not Changed

In Fig. 4.15 there is revealed what happens when 5-second-long library sounds are considered (right column) against one-second-long library sounds (left column). Note that the one-second library sounds correspond to the first second of the 5-second in the right column. The results are similar since most of the detected frames from the 5-second library are the frames from the first second. Tests where o-bank and e-bank fit (i) the most, (ii) less and (iii) the least are shown in top, middle and the bottom row, respectively. See in Fig. 4.15.

4.8.4 Change in Length of Observed Signal While Other Parameters Not Changed

In Fig. 4.16, the results can be seen when the Mozart subpart of SD #3, 500 frames long (left column), is tested against the whole SD #3 (right column). The result could be affected by the scaling-in-frequency matrix $\boldsymbol{D}_\phi$ when it is calculated from the shorter observed data against when the observed data are longer. The result is as such: the length of the observed data does not affect the estimation accuracy, it is the number of distinct notes that affects it. We also tried only observing 80-frame-long signal, and the result was the same approximately. It must be mentioned here, that $\boldsymbol{D}_\phi$ is normalized, see eq. (3.8). In Fig. 4.16, as in Subsection 4.8.4, the tests where o-bank and e-bank fit (i) the most, (ii) less and (iii) the least are shown in top, middle and the bottom row, respectively. All tests had 5-second-long library sounds.

4.8.5 Scaling in Frequency vs. No-scaling at All

In Fig. 4.17 it can be seen that with no scaling $\boldsymbol{D}_\phi$ (left column) the results are worse both in the SDRs and in the hits #2 than with scaling in frequency (right column). Moreover, the range of r_ω narrows without scaling and its maximum shifts to lower values. The tests were performed on the SD #2. In Fig. 4.17, as in Subsections 4.8.3, 4.8.4, the top, middle and the bottom row correspond to the most, less and the least e-bank and o-bank fit, respectively. All tests had 5-second-long library sounds.

4.8.6 Time and Frequency Scaling vs. Frequency Scaling Only

In Fig. 4.18, the reader can compare tests with scaling in frequency that were performed along with (left column) and without scaling in time (right column). The results almost do not differ, both in the curves of shape of SDR and in hits #2. Only in the last comparison (the least fit), we have different shapes of the result curves. The F-measure in the hits #2 has almost the same highest value regardless of scaling in time application. We conclude that the effect of the scaling in time cannot be assessed. The tests were performed on the SD #2. In Fig. 4.17, as in Subsections 4.8.3, 4.8.4, 4.8.5, the top, middle and the bottom row corresponds to the most, less and the least e-bank and o-bank fit, respectively. All tests had 5-second-long library sounds.

4.8.7 Exact Fit Tests

In order to prove the algorithm and model accuracy the exact fit tests were carried out. By the exact fit tests we mean an experimental setup where the o-bank and e-bank are identical in their length and content. In Fig. 4.19, it can be seen that the exact fit tests reach almost 100% accuracy in F-measure over a wide range of r_ω. Then we can see how at least a low selected r_ω contributes to a significant increase in accuracy both in the SDRs and in hits #1 and hits #2. And also, we can see that curves of hits #1 and hits #2 accuracy matches. It implies that there are no false-positives-within-the-sound. The exact fit tests were accomplished on SD #2 and SL #2.

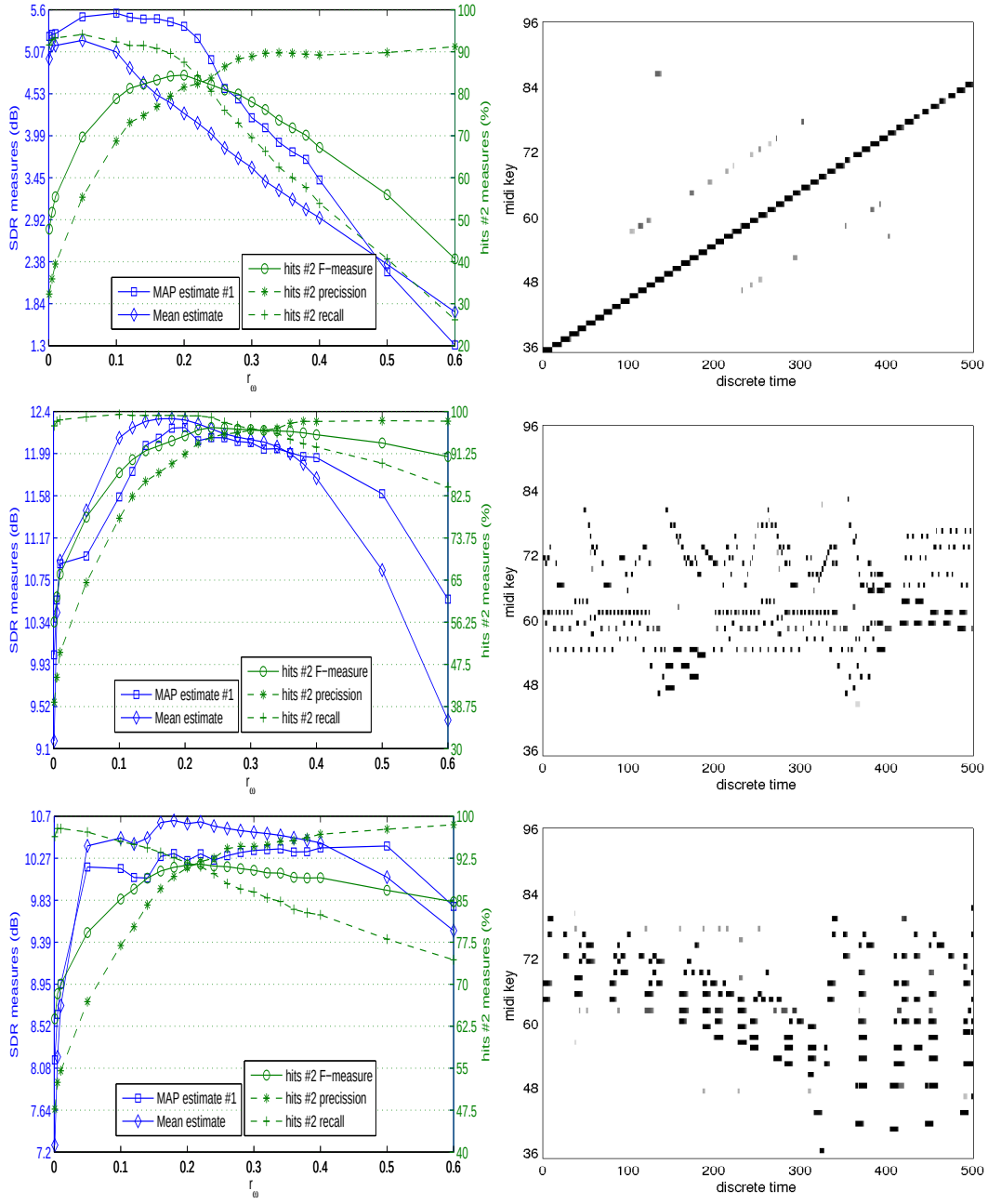


Figure 4.11: Testing sound libraries: o-bank – SL #1, e-bank – SL #2. The left column contains SDR and hit #2 measures for different r_ω whereas the right column is a piano-roll representation of $\boldsymbol{L}$ estimate for $r_\omega = 0.24$ from the left column. The first row: the note sequence. There are 50 tones from 61 on 500 length, the remaining of 11 tones are omitted; the second row: Mozart part from SD #3; the third row: Debussy part from SD #3.

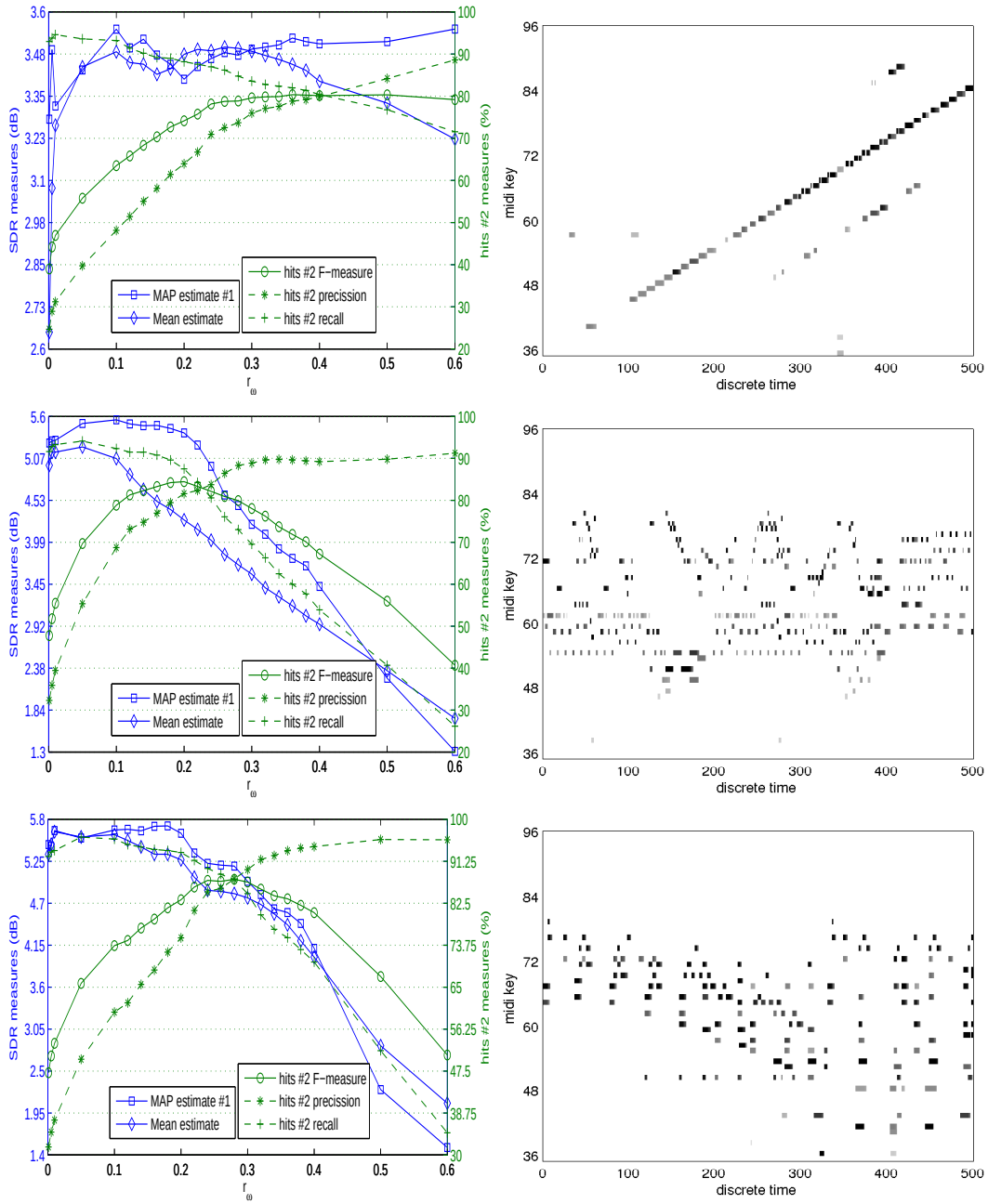


Figure 4.12: Testing sound libraries: o-bank – SL #3, e-bank – SL #2. The left column contains tests on SDR and hit #2 measures whereas the right column is a piano-roll representation of the test for $r_\omega = 0.24$ from the left column. The first row: the note sequence; the second row: Mozart part from SD #3; the third row: Debussy part from SD #3.

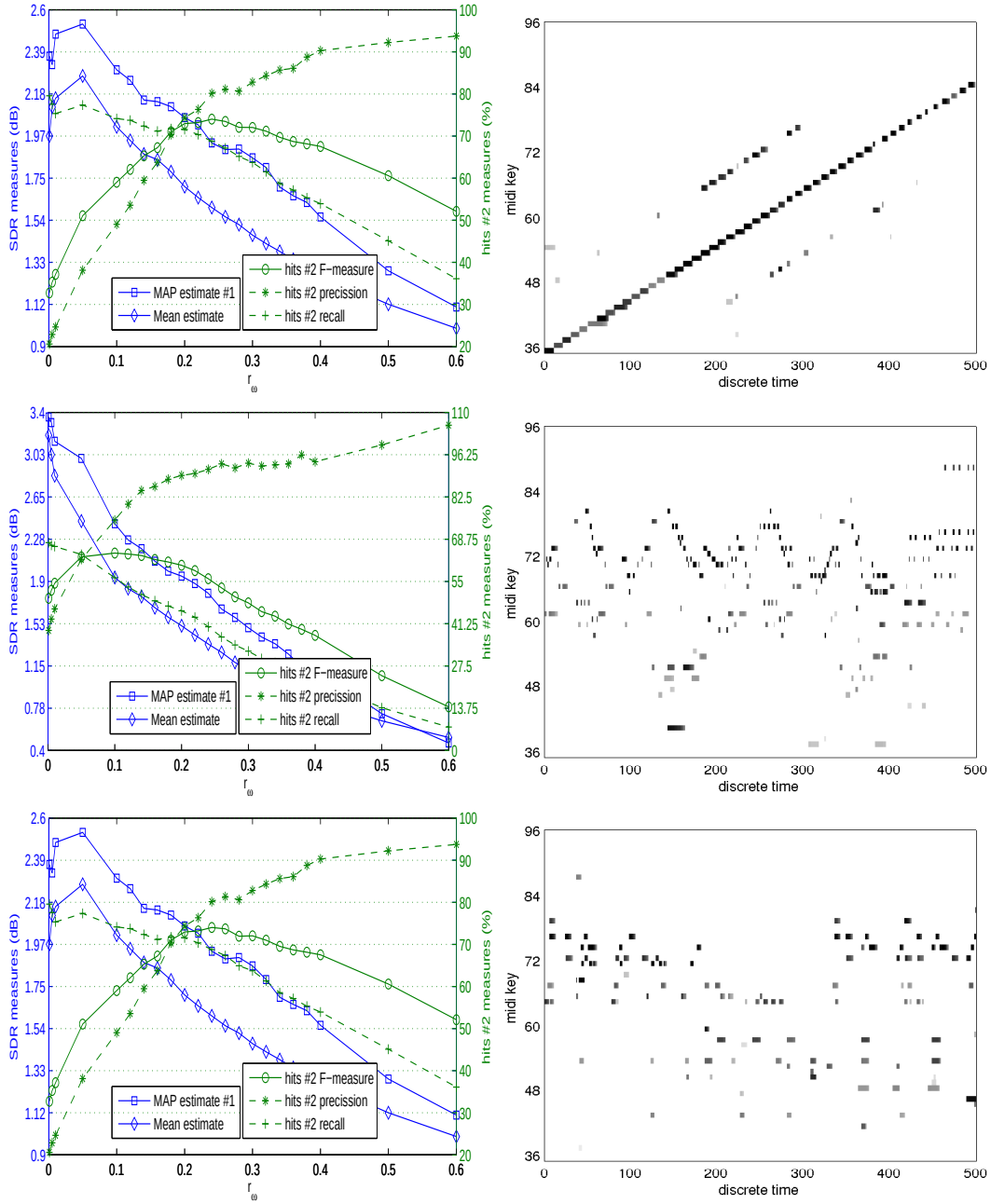


Figure 4.13: Testing sound libraries: o-bank – SL #4, e-bank – SL #2. The left column contains tests on SDR and hit #2 measures whereas the right column is a piano-roll representation of the test for $r_\omega = 0.20$ from the left column. The first row: the note sequence; the second row: Mozart part from SD #3; the third row: Debussy part from SD #3.

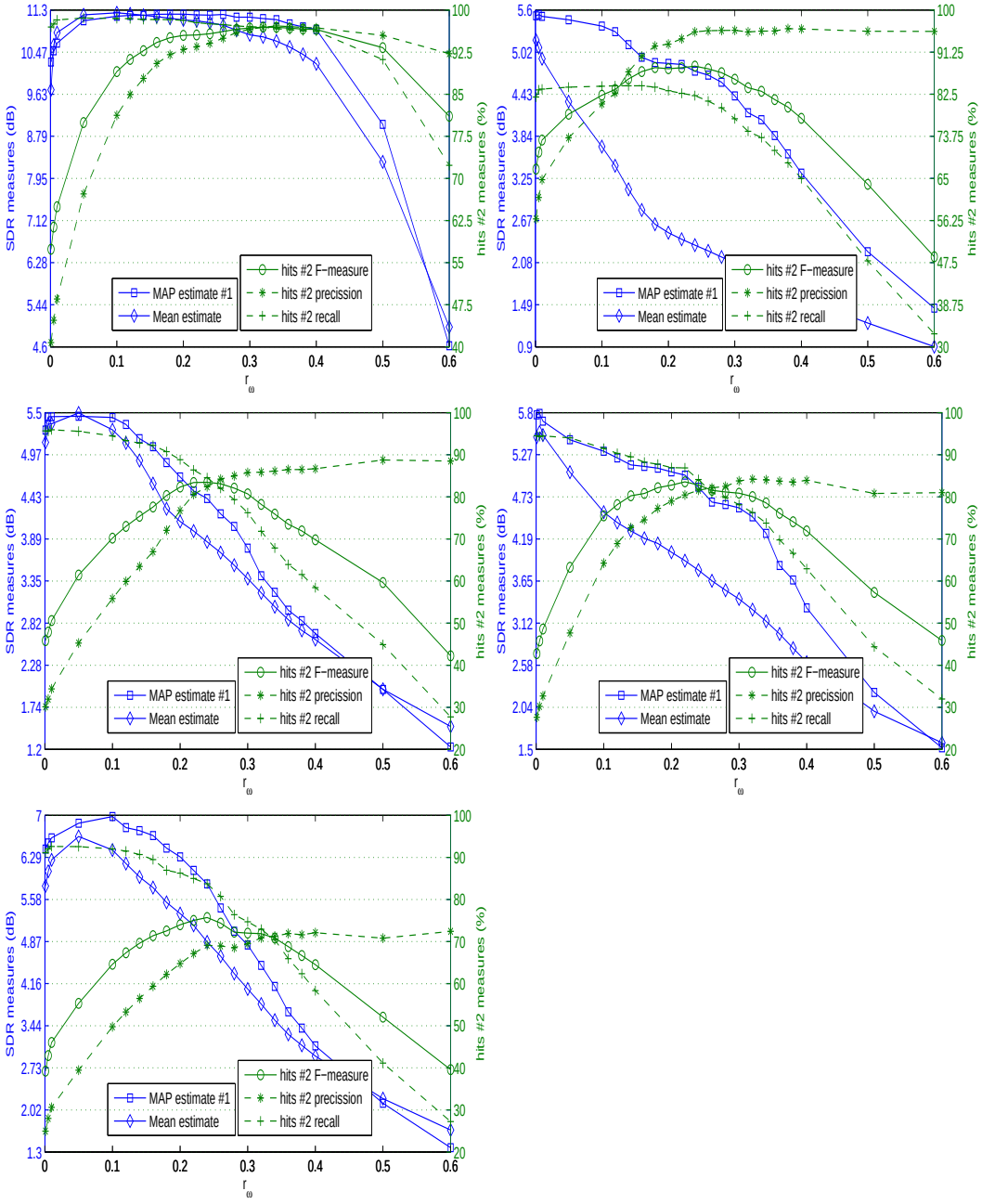


Figure 4.14: Top-left: o-bank – sound library (SL) #2, e-bank – SL #1 (forte vs. mezzo-forte piano); middle-left: o-bank – SL #3, e-bank: SL #2; bottom-left: o-bank – SL #2, e-bank – SL #3; top-right: o-bank – SL #2, e-bank – SL #2 – one-second shifted; middle-right – o-bank: SL #4, e-bank – SL #2.

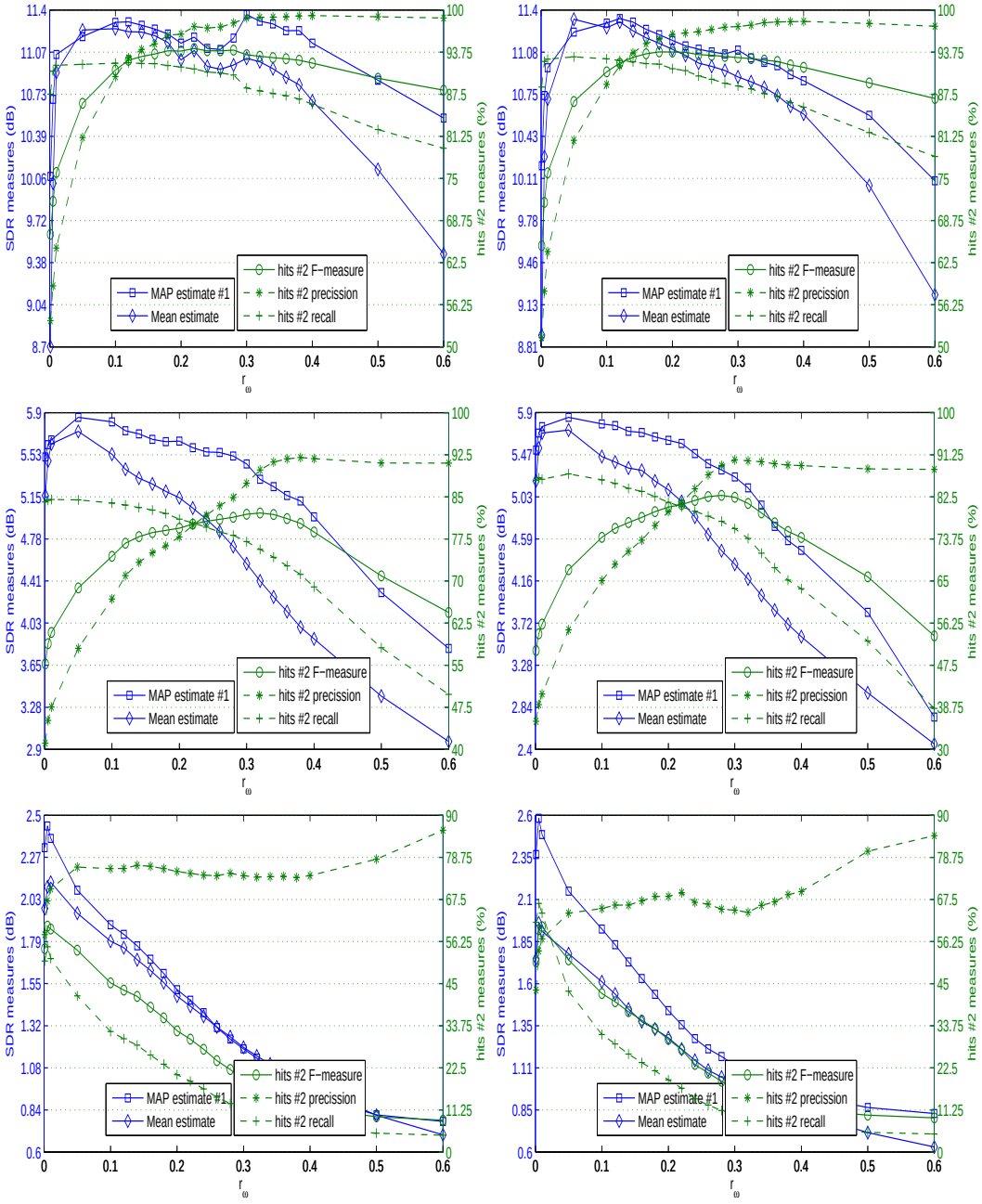


Figure 4.15: Top-left: SD #3, o-bank – SL #2, e-bank – SL #1, one-second-long library sounds; top-right: SD – #3, o-bank – SL #2, e-bank – SL #1, 5-second-long library sounds; middle-left: SD #3, o-bank – SL #3, e-bank – SL #2, one-second-long library sounds; middle-right: SD #3, o-bank – SL #3, e-bank – SL #2, 5-second-long library sounds; bottom-left: SD – #3, o-bank – SL #4, e-bank – SL #2, one-second-long library sounds; bottom-right: SD #3, o-bank – SL #4, e-bank – SL #2, 5-second-long library sounds.

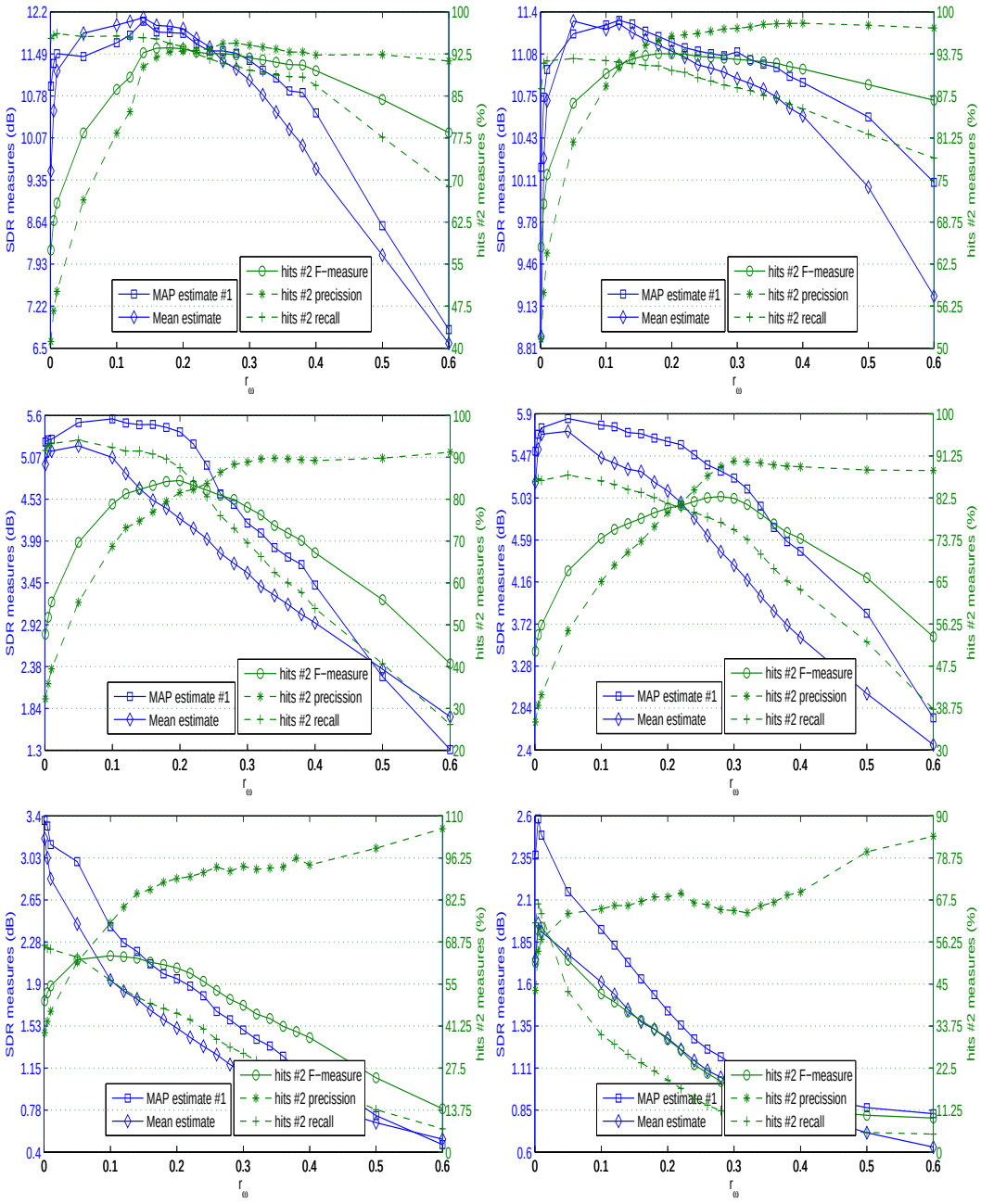


Figure 4.16: Top-left: the Mozart subpart of SD #3, o-bank – SL #2, e-bank – SL #1; top-right: SD #3 (whole), o-bank – SL #2, e-bank – SL #1; middle-left: the Mozart subpart of SD #3, o-bank – SL #3, e-bank – SL #2; middle-right: SD #3 (whole), o-bank – SL #3, e-bank – SL #2; bottom-left: the Mozart subpart of SD #3, o-bank – SL #4, e-bank – SL #2; bottom-right: the Mozart subpart of SD #3, o-bank – SL #4, e-bank – SL #2.

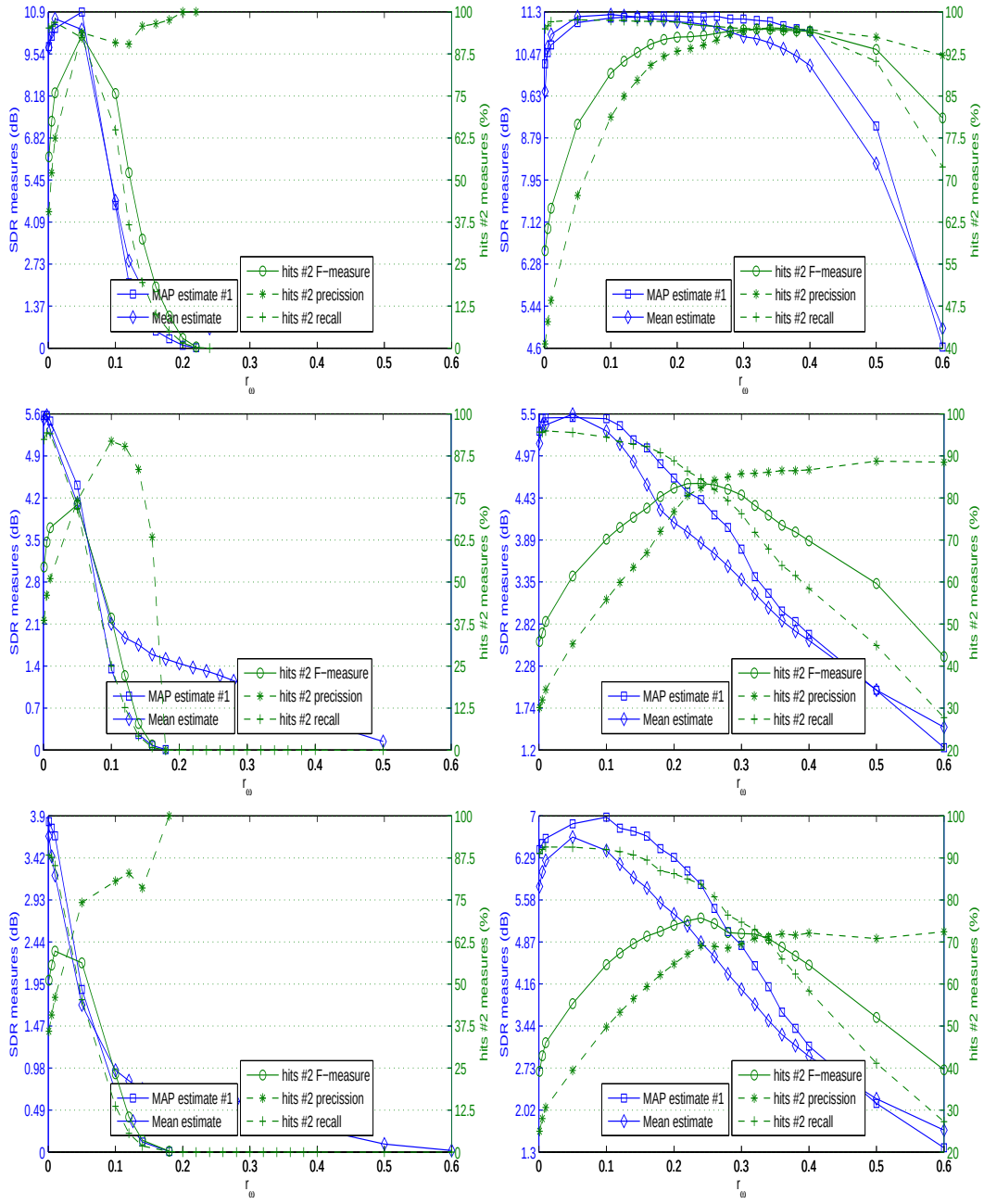


Figure 4.17: All tests were performed on SD #2. Top-left: o-bank – SL #2, e-bank – SL #1, no scaling; top-right: o-bank – SL #2, e-bank – SL #1, scaling in frequency; middle-left: o-bank – SL #3, e-bank – SL #2, no scaling; middle-right: o-bank – SL #3, e-bank – SL #2, scaling in frequency; bottom-left: o-bank – SL #4, e-bank – SL #2, no scaling; bottom-right: o-bank – SL #4, e-bank – SL #2, scaling in frequency.

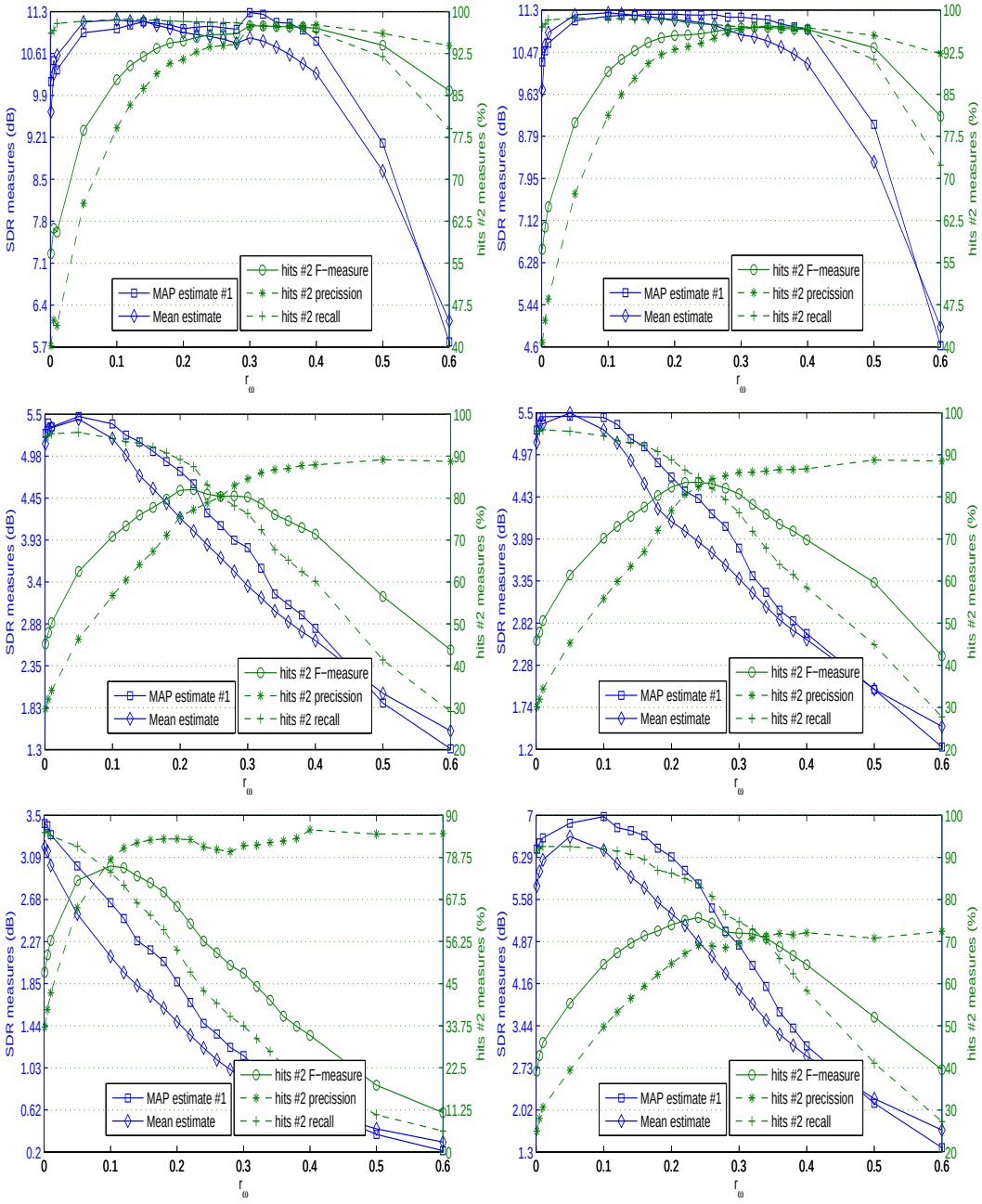


Figure 4.18: All tests were performed on SD #2. Top-left: o-bank – SL #2, e-bank – SL #1, both scaling; top-right: o-bank – SL #2, e-bank – SL #1, scaling in frequency; middle-left: o-bank – SL #3, e-bank – SL #2, both scaling; middle-right: o-bank – SL #3, e-bank – SL #2, scaling in frequency; bottom-left: o-bank – SL #4, e-bank – SL #2, both scaling; bottom-right: o-bank – SL #4, e-bank – SL #2, scaling in frequency.

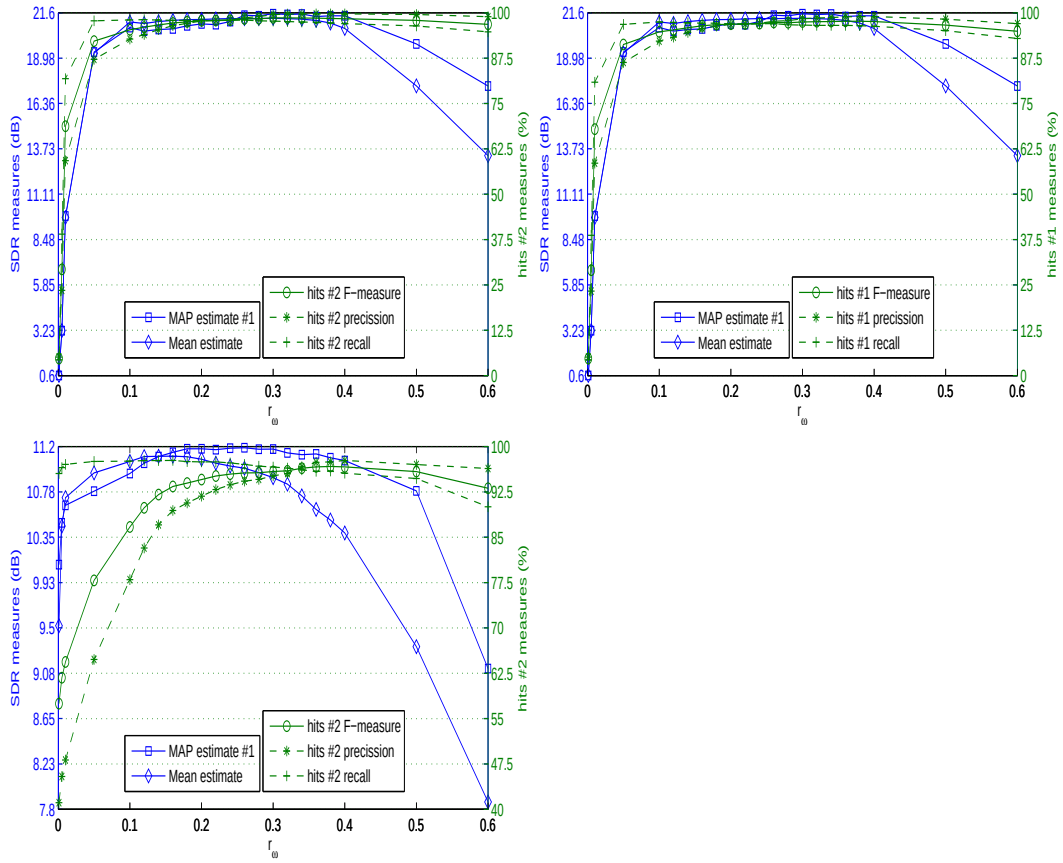


Figure 4.19: Top-left: SDR measures and hits #2 measures, top-right: SDR measures and hits #1 measures, bottom: comparison with non-exact fit test – here we operate with one-second SL #1 in the e-bank (when the o-bank is from SL #2).

4.8.8 Summary on Tests without Estimation of Amplitudes

In this section, we kept the amplitudes fixed at the best average estimate value. In order to see what is the strength of the influence by a parameter, we need (i) one test when the parameter is held fixed on the ground-truth or the best estimate and (ii) a set of tests of the parameter realizations. E.g., the ground truth for selection of $\boldsymbol{F}^{est}$ is $\boldsymbol{F}^{obs}$, we carried out tests for a several $\boldsymbol{F}^{est}$. Sometimes, the ground-truth or the best estimate is not known beforehand and we can only test the realizations (ii) as, e.g., in the case of the covariance structure of noise ω or, e.g., when the parameter "application

of scaling" has its support on values "yes", "no".

- The covariance structure of noise ω was estimated as one scalar for each element of the observation covariance matrix. Since most of the frequency bins in the higher frequencies of the observed data have very low magnitudes (even after scaling of the data), those below the threshold need to be excluded from the noise scalar estimation. A single threshold could not be set for all tests, the shape of the hit measures on the threshold support yields a concave curve in most cases, whereas the curve for the SDRs was usually monotonically decreasing.

- The estimation of labels is significantly affected by the selection of the sound libraries for estimation, by the number of sounds in the sound library, by the number of distinct sounds in the input signal, by the values of the transition matrix and by the application of scaling in frequency. It is insignificantly influenced by longer library sounds when the sound intrinsic subsegments do not match other sounds from the library. It is not influenced by the length of the observation signal. The effect of scaling in time was not conclusive. If the input recording is combined from piano sounds played "forte" and the sound library is collected by piano sounds played "mezzo-forte", the results in hits of the F-measure get over 90% and in the SDR exceed 10 dB in all simulated data sets. However, when the input recording was made from a different type of piano (e.g., the electric piano) then the richer and more polyphonic recording produced significantly worse results, whose F-measure value ranged between 60 – 70%.

4.9 Tests of Estimation with Amplitudes

The estimation from the model with amplitudes was tested with two distinct settings of the distributions on amplitudes. The first (see in Subsections 4.9.1, 4.9.2) sets up that there is no relation between amplitudes and, second (see in Subsection 4.9.3), that all amplitudes are the same approximately. The tests were performed on SD #3 with sound libraries having 5-second-long sounds.

4.9.1 Investigation of Sparsity Constraint on Amplitudes without Any Defined Relation among Each Other

The requirement is characterized by a distribution of amplitudes $a_s \sim \mathcal{N}(\mu_{hyp,a,0} = 0, \sigma_{a,0})$, i.e., $\mu_{hyp,a,0}$ being fixed at zero and $\sigma_{a,0}$ denoting the

level of sparsity constraint and representing the investigated parameter. In Fig. 4.20, the x-axis represents values of $\sigma_{a,0} = \epsilon$. A reader may notice that at the variance $\sigma_{a,0} = 0.002$ the sparsity constraint proves itself in a slight accuracy improvement in hits #2. If the o-bank and the e-bank correspond enough (top figure from the triple) then ϵ_{optim} for the hits #2 curves corresponds to ϵ_{optim} in the SDR curves. In case of a bigger difference between o-bank and e-bank (bottom graph in Fig. 4.20), the variance ϵ_{optim} for hits #2 curves is shifted to a higher value and is not matched with the ϵ_{optim} for SDR curves. The threshold for ω calculation r_ω was set to 0.2 for all tests of dependency on ϵ in Fig. 4.20.

4.9.2 Amplitudes without Any Defined Relation between Each Other – Tests with Optimal ϵ

We consider $a_s \sim \mathcal{N}(\mu_{hyp,a,0} = 0, \sigma_{a,0} = \epsilon_{optim})$, $\epsilon_{optim} = 0.002$ for all tests. The tests of the most, less and the least o-bank and e-bank fit are denoted in Fig. 4.21 (the most) and 4.22 (less and the least), respectively. The threshold for ω calculation r_ω was set to 0.2 for all tests. After 10 cycles of convergence, the F-measure in labels F_2 reads values 93%, 80%, 63% for the most, less and the least fit between o-bank and e-bank, respectively. Their standard deviations in amplitudes correspond to 0.17, 0.23, 0.22.

4.9.3 All Amplitudes Have the Same Value a Approximately

Let us consider such settings: $\mu_{hyp,a,0}$ is estimated, $\sigma_{a,0}$ is positive and close to zero[4] and $\sigma_{\mu,0} \to \infty$. This setup corresponds to the requirement when all amplitudes $\boldsymbol{a}$ have the same value and the number of variables for amplitudes is decreased from S to one. In Fig. 4.23 – 4.28 there are tests of cases in which $\sigma_{a,0} = 10^{-6}$ and $\sigma_{a,0} = 10^{-5}$. The value of $\sigma_{a,0}$ close to zero ensures that the amplitudes will be forced to move together in their value while iterating in Algorithm 3. It can be seen from the figures that the tests with $\sigma_{a,0} = 10^{-5}$ converge faster in $\mu_{hyp,a,0}$ but the amplitudes differ more, whereas the tests with $\sigma_{a,0} = 10^{-6}$ converge slowly in $\mu_{hyp,a,0}$ but the differences among amplitudes are negligible. Due to the computation load and the provided time span of 24 hours for the program run itself, the maximum number of cycles was about one hundred. The tests of the most, less and the least fit in o-bank and e-bank are denoted in Fig. 4.23, 4.26 (the most), 4.24, 4.27 (less) and 4.25, 4.28 (the least), respectively. From the figures can be seen

[4]But not as close to attack not-a-number numeric representation.

that the common estimated amplitude for all library sounds is not the same in the tests and decreases from most to least fit in the banks. The tests were performed on SD #3 with sound libraries having 5-seconds long sounds.

In Fig. 4.29 there are results of the hit #2 and the SDRs of the last iteration of the convergence when $\sigma_{a,0} = 10^{-5}$ for the three "fit cases". After examination of the bottom image of the piano-roll in Fig. 4.29, one could observe that the result is really poor in this case. In fact the total F-measure for this case is 58%. It can be reasoned by the high recall value which increases the total F-measure. After convergence, the F-measure F_2 reads values 92.5%, 82.5%, 56% for Fig. 4.26, 4.27, 4.28, respectively.

The common estimated amplitude $\mu_{hyp,a,0}$ differs according to the o-bank and e-bank fit – if the o-bank and e-bank fit is the greatest, the estimated $\mu_{hyp,a,0}$ is of the highest value while if the o-bank and e-bank fit is the lowest, the estimated $\mu_{hyp,a,0}$ is the lowest, too. The estimated value of $\mu_{hyp,a,0}$ ranges between 0.55 to 0.82 while the tests with the fixed amplitude (in Section 4.8) were performed using the value 0.65. The difference between $\mu_{hyp,a,0} \doteq 0.55$ and $\mu_{hyp,a,0} \doteq 0.65$ in experiments with fixed value (see in Fig. 4.16 in its bottom-right image) does not have much impact on results.

The next observation is that the tests with estimation of the common amplitude (this Subsection) and the tests with 61 amplitudes (Subsection 4.9.2) do not significantly differ in results, that is, the increase of the number of unobserved variables by estimating of amplitudes did not significantly changed the total results. Only in the least suitable selection of the library for the estimation (the electric piano instead of the "classical" piano) we can see, that the common amplitude test yielded 7% worse result in hits #2 (F-measure) over the tests with amplitudes estimated for all library sounds. This can be explained by the fact that errors (i.e., unsuitability) in the estimation library are compensated by more free parameters in the model.

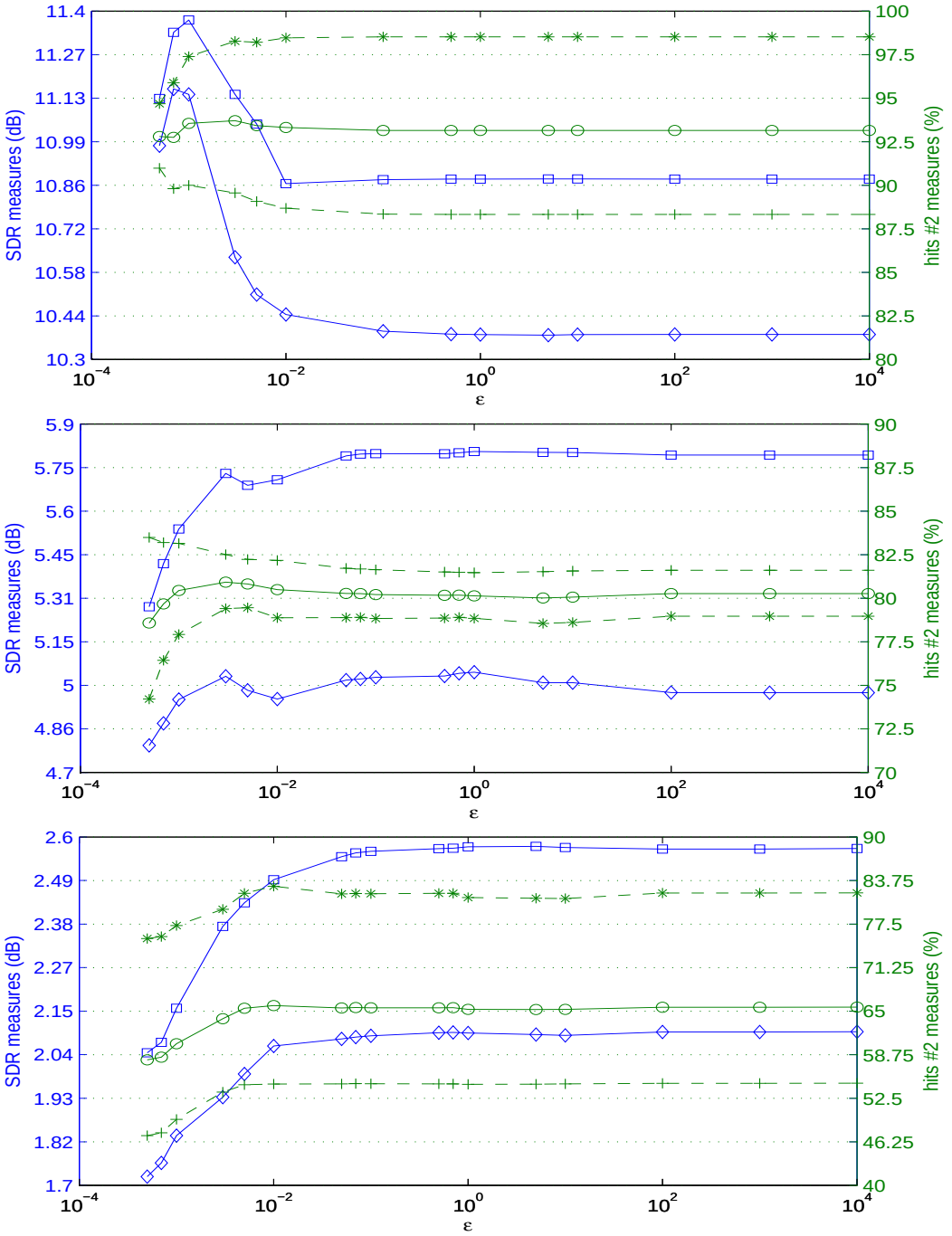


Figure 4.20: ϵ tests. Top: o-bank – sound library #2, e-bank – sound library #1; middle: o-bank – sound library #4, e-bank – sound library #2; o-bank – sound library #3, e-bank – sound library #2.

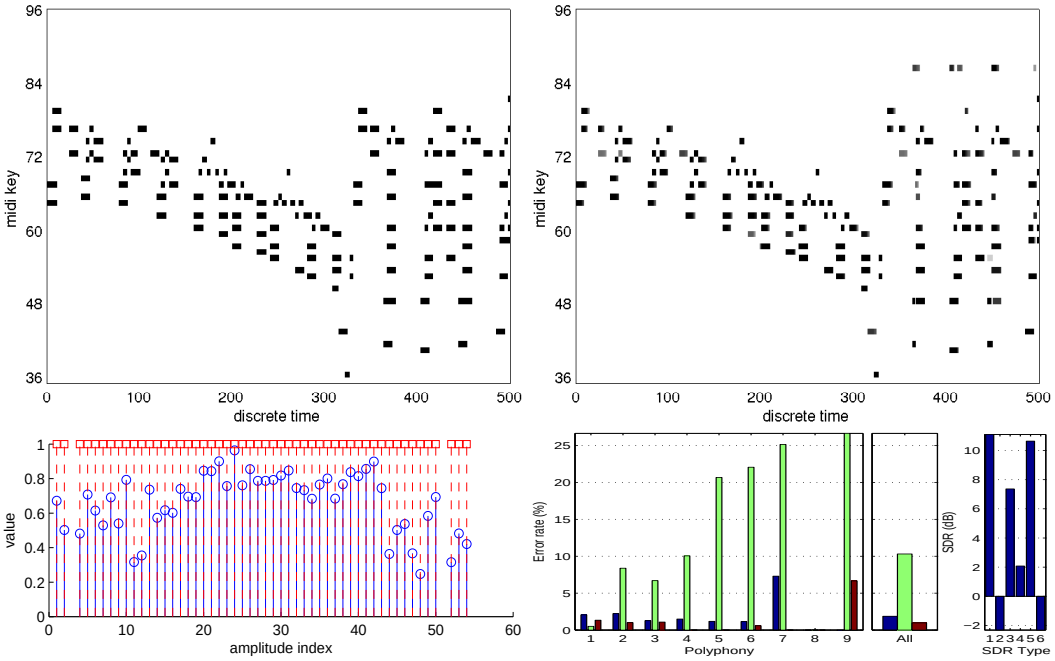


Figure 4.21: Top-left: the ground truth music excerpt from the testing data in the piano-roll; top-right: its transcription (disregarding amplitudes); bottom-left: circles – values of amplitudes of detected library sounds (e-bank) vs. squares – indices of present library sounds in the observed signal (o-bank); bottom-right: total results.

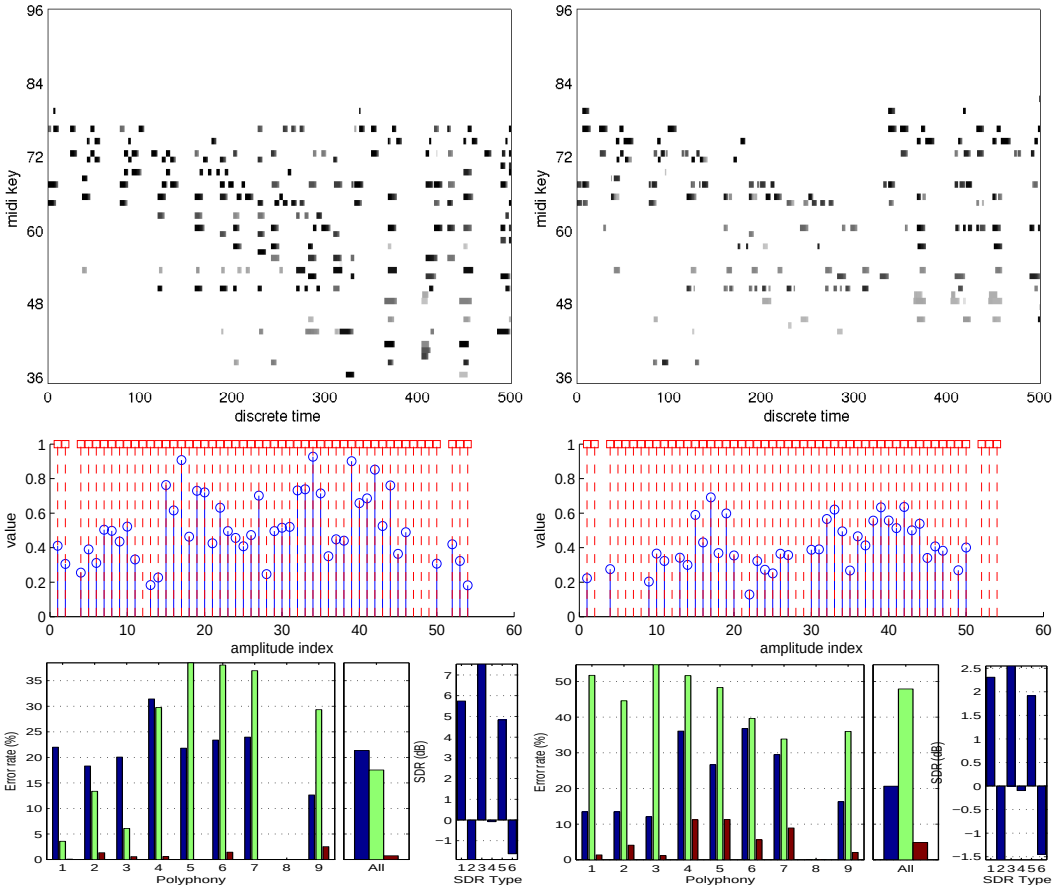


Figure 4.22: The less (left column) and the least (right column) fit in o-bank and e-bank.

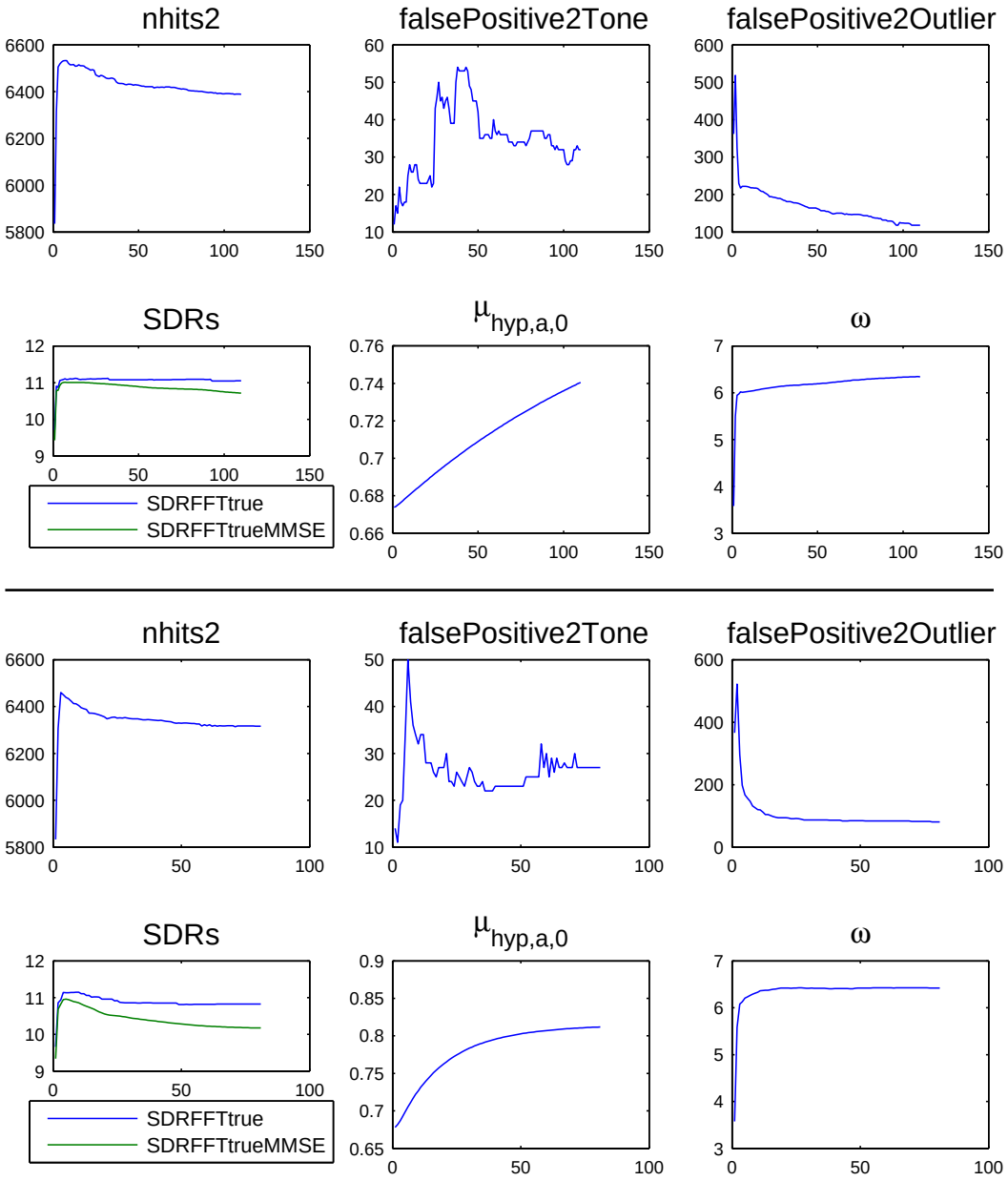


Figure 4.23: Convergence of quantities when amplitudes are forced to move together in their amplitude, the case when o-bank – SL #2, e-bank – SL #1. Upper: $\sigma_{a,0} = 10^{-6}$, lower: $\sigma_{a,0} = 10^{-5}$. X-axis: iteration cycle, y-axis: quantity value.

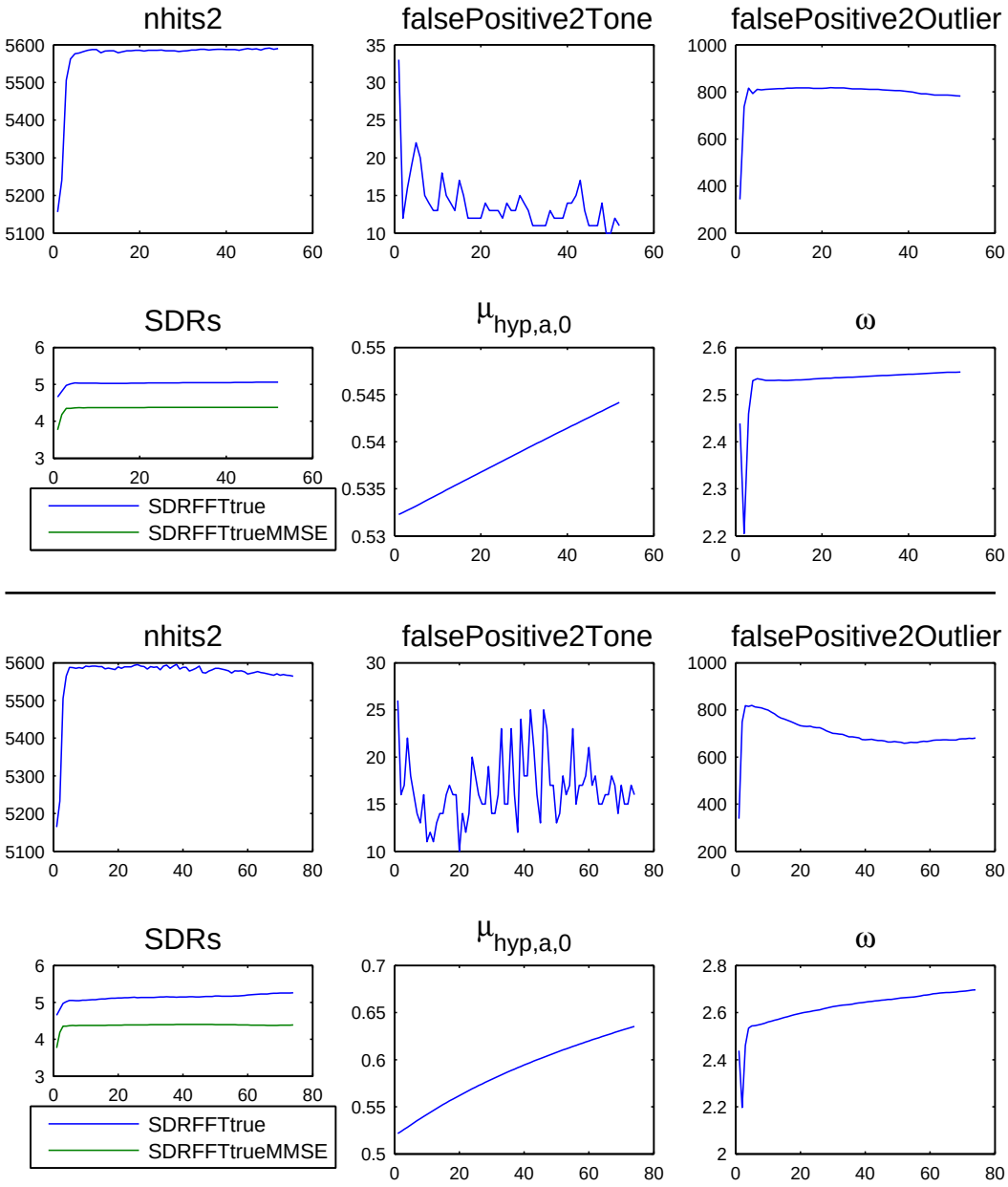


Figure 4.24: Convergence of quantities when amplitudes are forced to move together in their amplitude, the case when o-bank – SL #3, e-bank – SL #2. Upper: $\sigma_{a,0} = 10^{-6}$, lower: $\sigma_{a,0} = 10^{-5}$. X-axis: iteration cycle, y-axis: quantity value.

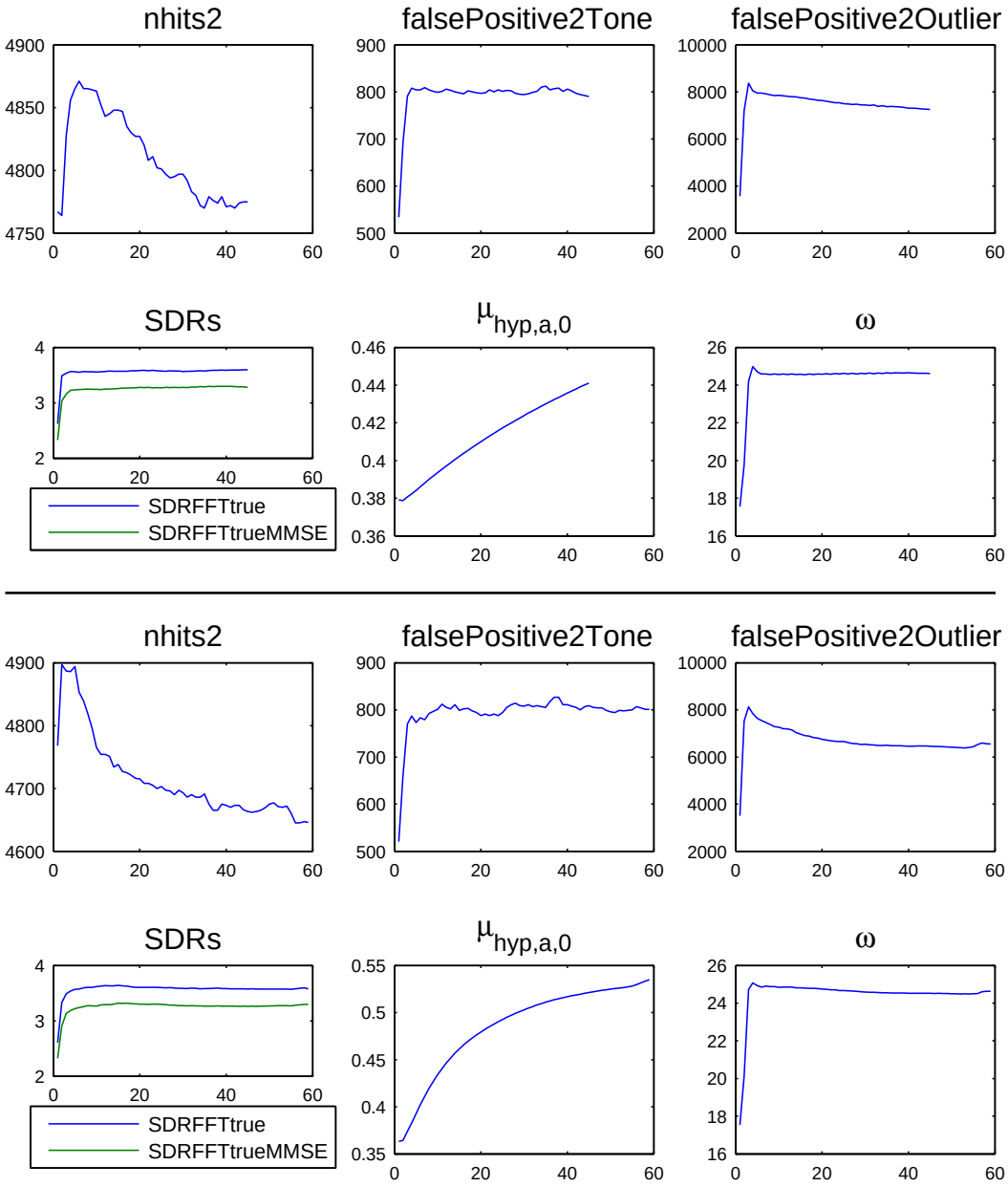


Figure 4.25: Convergence of quantities when amplitudes are forced to move together in their amplitude, the case when o-bank – SL #4, e-bank – SL #2. Upper: $\sigma_{a,0} = 10^{-6}$, lower: $\sigma_{a,0} = 10^{-5}$. X-axis: iteration cycle, y-axis: quantity value.

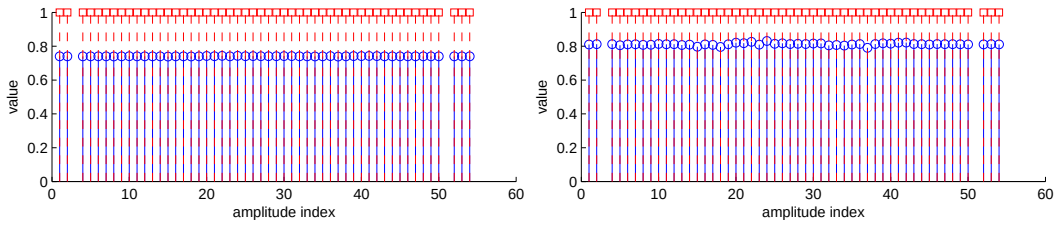


Figure 4.26: Last iteration of the convergence of amplitudes when they are forced to move together in their amplitude value, the case when o-bank – SL #2, e-bank – SL #1. Left: $\sigma_{a,0} = 10^{-6}$, right: $\sigma_{a,0} = 10^{-5}$.

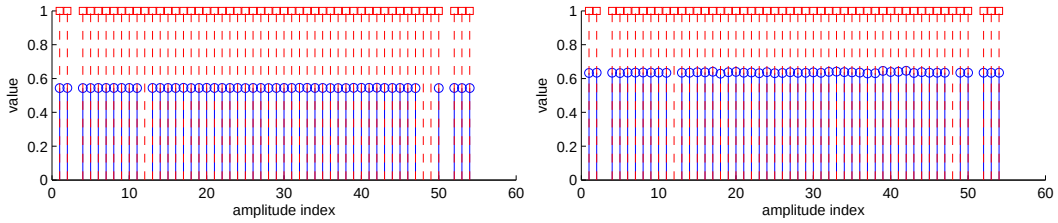


Figure 4.27: Last iteration of the convergence of amplitudes when they are forced to move together in their amplitude value, the case when o-bank – SL #3, e-bank – SL #2. Left: $\sigma_{a,0} = 10^{-6}$, right: $\sigma_{a,0} = 10^{-5}$.

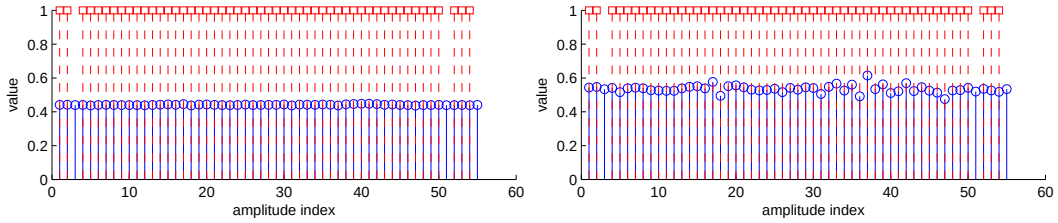


Figure 4.28: Last iteration of the convergence of amplitudes when they are forced to move together in their amplitude value, the case when o-bank – SL #4, e-bank – SL #2. Left: $\sigma_{a,0} = 10^{-6}$, right: $\sigma_{a,0} = 10^{-5}$.

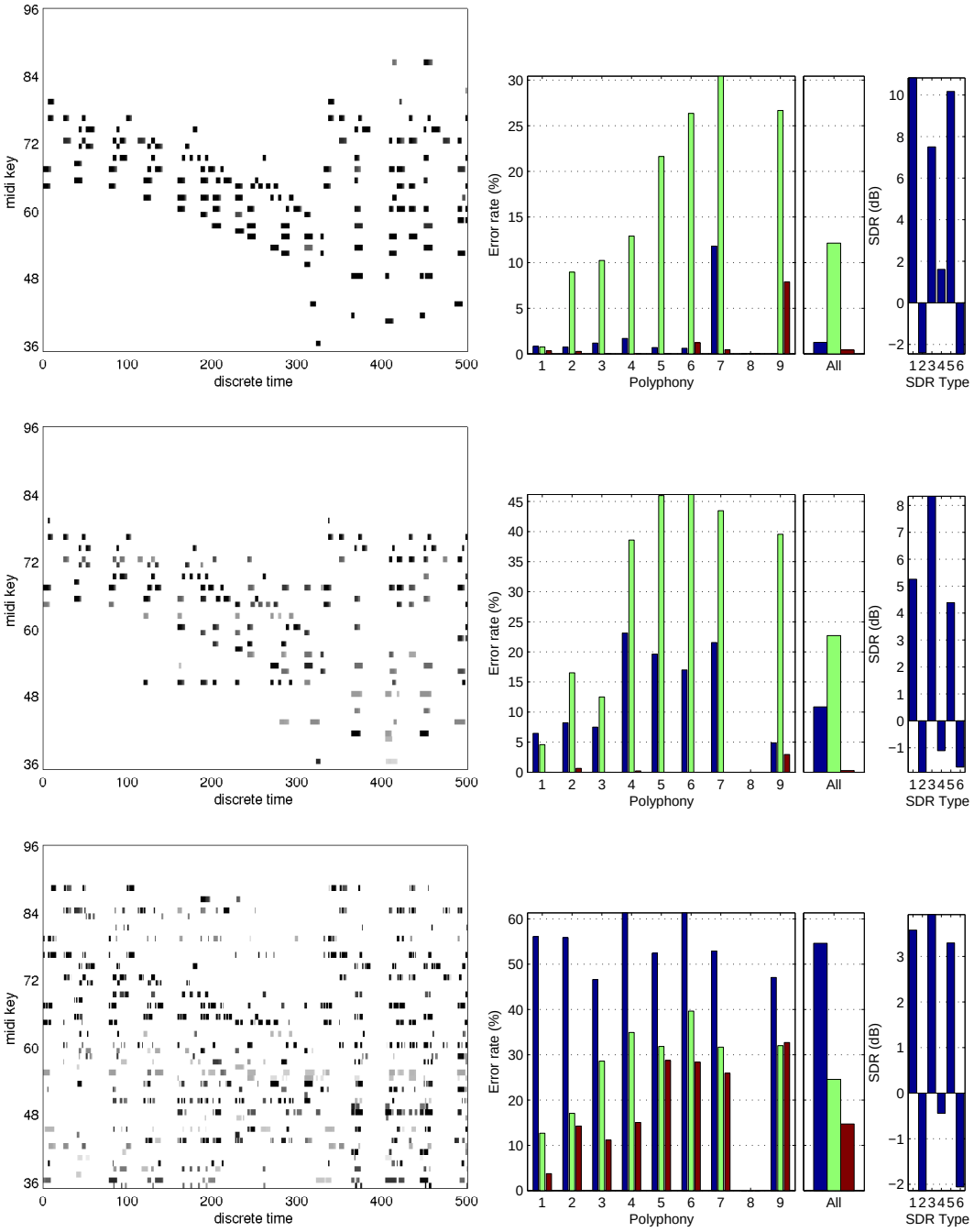


Figure 4.29: Last iteration of the convergence when amplitudes are forced to move together. Top: o-bank – SL #2, e-bank – SL #1, corresponding amplitude image in Fig. 4.26; middle: o-bank – SL #3, e-bank – SL #2, corresponding amplitude image in Fig. 4.27; bottom: o-bank – SL #4, e-bank – SL #2, corresponding amplitude image in Fig. 4.28.

4.9.4 Summary on Tests with Estimation of Amplitudes

The amplitude is estimated one for each library sound and it does not change over time. By the settings of prior distribution on amplitudes two cases were tested:

- 61 amplitudes are independent among each other and they are pushed to zero by a sparsity constraint. Tests revealed that L_2 norm sparsity constraint improves slightly the accuracy. Having the best average value for the sparsity constraint, we conclude that: even if the sound library for the observed signal is selected well, the amplitudes may vary a lot. In the case of the best fit of the sound library for the observed signal, we obtained the standard deviation for the amplitudes of 0.17.
- Amplitudes are tied to each other by their variance (hyperparameter): one common amplitude estimated, no sparsity constraint. The more similar are the library for estimation and the library, the observed signal was created from, the higher is the common estimated amplitude. Also it holds, that the more the amplitudes are tied to each other, the slower is the convergence of the VB algorithm. The decrease of the number of unobserved variables by estimating of one common amplitude instead of 61 (one for each library sound) did not significantly change the total results (hits #2, SDR), when the library of sounds for estimation was of the same piano-type sound as the sound library used on creation of the observed signal.

4.10 Comparison to Multiple Fundamental Frequency Estimation State-of-the-art

The multiple fundamental frequency estimation is the optional functionality provided by our model of the inverse music sequencer. The comparison follows from the overview of approaches in Section 1.3 and the multiple-fundamental frequency state-of-the-art in Section 1.6. In Table 4.1, the approaches are presented with their accuracies. In that table, our approach is represented by a test result from the sparsity test with estimation of amplitudes in Fig. 4.20, upper picture. In our compared approach, the o-bank and e-bank are distinctly recorded but very close libraries and the sparsity tuning parameter $\sigma_{a,0}$ is set to the optimum. In all the methods to compare, we selected the most comparable tests and their settings, all percentages correspond to the tests on simulated data. The abbreviations used in the table

are explained in List of Abbreviations at the beginning of the thesis. We conclude that:

- The most similar approach to ours is of Abdallah and Plumbley [43]. Besides the signal model and the parameter estimation method(s), ours and [43] differ in the prior source knowledge selection. Whereas in [43], the data sources are calculated in advance from the observed data itself and the meaningful spectral vectors are selected by hand, our approach needs to select a suitable sound library before starting the estimation algorithm, no human step into is necessary.

- If it is possible we provide the R-index evaluation (1.12) for the methods. It is clear to see that the proposed solution compete well with the state-of-the-art.

Authors; approach; Method – description	Source specific/ model based/ statistical	Freq- uency: resol. / maxim.	Testing data					Evaluation measures of accuracy; on	Accuracy in hits
			instru- ments: real (r), synth (s)	length	distinct chords vs. song	average polyph.	number of events		
Our approach; VB method	Y / Y / Y	10Hz / 6 kHz	piano (r)	5.25min	song	2.25	tones: 1566 frames: 7219	(i) F-measure, (ii) R-index (1.12); on frames	(i)93.7% (ii)94.0%
Abdallah & Plumbley (2006)[43], mostly gradient based method	Y / Y / Y	21Hz / 5.5 kHz	harpsi- chord (unsp.)	unsp.	song	unsp.	tones: 4684 frames: unsp.	(i) correctly det., (ii) false-posit., (iii) R-index (1.12); on tones	(i)94.3% (ii)2%, (iii) 96.2%
Kashino (1995) [4]; ASA, Bayesian Networks	Y / N / Y	unsp. / unsp.	piano, flute, violin (s)	unsp.	unsp.	3	tones: unsp. frames: unsp.	R-index (1.12) for (i) 2 tones in octave, (ii) 2 in fifth, (iii) no relation	(i)79%, (ii)71%, (iii)67%
Klapuri (2006) [16]; Human audit. system model	N / N / N	10Hz / 6kHz	2842 samples of 32 instr. (s)	NM	distinct chords	2, 4	tones: 4000 frames: 4000	correctly det. for (i) poly. 2, (ii) poly. 4;	(i)94% , (ii)88%
Vincent (2008) [41]; NMF	N / Y / N	unsp.	Disk- lavier piano (unsp.)	21.5min	song	2.1	tones: 9489 frames: unsp.	F-measure	87%
Marolt (2001) [15]; Human audit. system model, neural networks with features from adaptive oscillators	Y / N / N	unsp. / 6kHz	piano (s)	unsp.	song	3.0	tones: 12624 frames: unsp.	(i) correctly det., (ii) false-posit., (iii) R-index (1.12); on tones	(i)90%, (ii)9%, (iii)90.5%

Table 4.1: Comparison to multiple fundamental frequency estimation approaches

Chapter 5

Conclusions

We have presented a model for polyphonic music signal and an approximate algorithm for estimation of its parameters based on the Variational Bayes approximation. The model is based on the operation of a music sequencer and it is designed to operate with general music sounds. The algorithm is capable of identifying sounds and their amplitudes that do not change over time in the input polyphonic music signal. Only sounds within a given sound library can be identified. Additionally, unlike the state-of-the-art, the model allows identification of arbitrary subparts of the library sounds, not only the sound as a whole. We term this feature the "modification". The model can be extended by other modifications, e.g., the pitch-shift of a library sound. We have devised an evaluation method and applied it along with other standard methods in the algorithm evaluation. Simulated data, prepared by assigning real and synthesized piano sounds to classical music pieces in MIDI format, was used for the evaluation.

5.1 Contributions

- A scenario for the complete automatic music transcription [1] following the idea of an inverse music sequencer was presented. It allows working with a bank of drum and harmonic music sounds as a memory base (the "sound library for estimation"). Musical sounds in the observed audio signal are matched not only with the whole library sounds but also with their subparts.

- A probabilistic model containing unobserved variables of noise, amplitudes, labels (presences of sound segments – frames) and true inner library sounds was designed. The Variational Bayes technique was utilized to reduce the model equations and provide the algorithms for

estimation of unobserved variables, in particular Algorithm 3 and its modifications Algorithm 5 and Algorithm 6. It was shown that in the estimation of labels in our model the variational Bayes approach outperforms our previous estimation by the extended Kalman filter in discrimination of the labels.

- Evaluation methods reflecting measures of hits in labels and sound-to-distortion values were proposed. One of them, "hit mesure #2", was designed by the author and resulted in Algorithm 7. The methods were then applied in the evaluation of the proposed algorithms.

- The estimation of unobserved variables:

  - **Sound library:** its distribution is determined by the sound library for estimation. Considering such distribution, our tested model did not prove that 1/ the proposed distribution shape or 2/ greater length of the observed data can lead to improved results. Thus, in the further experiments, the sound library was considered to be a fixed parameter of the model, not a distribution[1].

  - **Covariance structure of noise:** it was estimated as a scalar for each element of the observation covariance matrix. Since most of the frequency bins in the higher frequencies of the observed data have very low magnitudes (even after scaling of the data), those below the threshold need to be excluded from the noise scalar estimation. A single threshold could not be set for all tests.

  - **Labels:** amplitudes are held fixed close to their average ground truth value. The estimation of labels is significantly affected by the selection of the sound libraries for estimation, by the number of sounds in the sound library, by the number of distinct sounds in the input signal, by the values of the transition matrix and by the application of scaling in frequency. It is insignificantly influenced by longer library sounds when the sound intrinsic subsegments do not match other sounds from the library. It is not influenced by the length of the observation signal. The effect of scaling in time was not conclusive. If the input recording is combined from piano sounds played "forte" and the sound library is collected by piano sounds played "mezzo-forte", the results in hits of the F-measure get over 90% and in the SDR exceed 10 dB in all simulated data

[1] Note that the fixed parameter can be expressed as a distribution too, see (3.40).

sets. However, when the input recording was made from a different type of piano (e.g., the electric piano) then the richer and more polyphonic recording produced significantly worse results, whose F-measure value ranged between 60 – 70%.

- **Labels with amplitudes:** the amplitude is estimated one for each library sound and it does not change over time. Having the same type of piano instrument in the sound library as in the observed signal, the tests with and without estimation of amplitudes do not significantly differ in results, that is, the increase of the number of unobserved variables by estimating of amplitudes did not significantly change the total results. By the settings of prior distribution on amplitudes two cases were tested:
  - **amplitudes are mutually independent**: they are pushed to zero by a sparsity constraint. Tests revealed that L_2 norm sparsity constraint improves slightly the accuracy. Having the best average value for the sparsity constraint, we conclude that even in the well selected sound library for the observed signal the amplitudes may vary a lot. In the case of the best fit of the sound library for the observed signal, we obtained the standard deviation for the amplitudes of 0.17 when the ground truth amplitudes were all equal to one common amplitude of a value between 0.5 and 1.0.
  - **amplitudes are tied to each other**: this is accomplished by setting hyperparameters for amplitudes. The observed signal was prepared from a sound library which is different from the sound library used for the estimation. The more similar these two libraries are, the higher the common estimated amplitude is. The algorithm exhibited faster convergence when the tying by the variance hyperparameter was not significant.

- The multi-pitch detection represents a way how to compare the performance in accuracy of our proposed method. The results show that when the inverse music sequencer is set up for memory-based multi-pitch detection and the library of sounds is selected suitably, we obtain over 90% correctly detected frames, which is also the level of the state-of-the-art of multiple fundamental frequency recognizers.

5.2 Future Work

- To extend the sound libraries by non-harmonic sounds, e.g., drums. To try to combine the tested sound library from sounds of instruments occurring in Western tonal music. Given this library, to create simulated data representing a popular song. In the first place, the tests should be performed only with estimation of labels, the amplitudes and library sounds would be held fixed.

- Estimation of the sound library by the three enhancements proposed in Section 4.1.

- Calculation of amplitudes that are changing over time: calculation of the VB algorithm for amplitudes and labels within a moving window, see in Subsection 3.4.

- Computational load reduction: since the whole observed signal is for disposal before the estimation is carried out, it is possible to precalculate suitable sounds for the library, or libraries, in the case of processing window by window (see in Section 3.4), in order to decrease the number of sounds in the library.

- To extend the set of modifications of sounds in the library, i.e., besides the model with truncation of sounds, we could consider the pitch-shift of the library sounds.

Bibliography

[1] M. Davy and A. Klapuri, eds., *Signal Processing Methods For Music Transcription*. Springer, 2006.

[2] M. M. Association, *Complete MIDI 1.0 Detailed Specification*, 1999/2008.

[3] E. S. Field, ed., *Beyond Midi: the handbook of musical codes*. London: MIT Press, 1997.

[4] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka, "Application of bayesian probability network to music scene analysis," in *Working Notes of IJCAI Workshop of Computational Auditory Scene Analysis (IJCAI-CASA)*, p. 52, Aug. 1995.

[5] M. Goto, "Music scene description project: Toward audio-based real-time music understanding," in *4th International Conference on Music Information Retrieval*, (Baltimore, USA), Oct. 2003.

[6] M. Ryynänen and A. Klapuri, "Transcription of the singing melody in polyphonic music," in *7th International Conference on Music Information Retrieval*, (Victoria, Canada), Oct. 2006.

[7] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, pp. 257–286, 1989.

[8] T. F. Quatieri, *Discrete-Time Speech Signal Processing: Principles and Practice*. Prentice Hall, Nov. 2001.

[9] S. Arberet, A. Ozerov, F. Bimbot, and R. Gribonval, "A Tractable Framework for Estimating and Combining Spectral Source Models for Audio Source Separation," Research Report RR-7556, INRIA, Mar. 2011.

[10] N. Duong, Q. K., E. Vincent, and R. Gribonval, "Under-Determined Reverberant Audio Source Separation Using Local Observed Covariance and Auditory-Motivated Time-Frequency Representation," in *Latent Variable Analysis and Signal Separation, 9th International Conference on* (V. Vigneron, V. Zarzoso, E. Moreau, R. Gribonval, and E. Vincent, eds.), vol. 6365 of *Lecture Notes on Computer Science*, (Saint-Malo, France), pp. 73–80, Springer, Sept. 2010.

[11] E. Vincent, S. Arberet, and R. Gribonval, "Underdetermined instantaneous audio source separation via local gaussian modeling," in *Proceedings of the 8th International Conference on Independent Component Analysis and Signal Separation*, ICA '09, (Berlin, Heidelberg), pp. 775–782, Springer-Verlag, 2009.

[12] S. Bregman, A., ed., *Auditory Scene Analysis*. The MIT Press, 1990.

[13] M. J. Hewitt and R. Meddis, "An evaluation of eight computer models of mammalian inner hair-cell function," *Acoustical Society of America*, vol. 90, pp. 904–917, 1991.

[14] A. Klapuri, *Signal Processing Methods for the Automatic Transcription of Music*. PhD thesis, Tampere University of Technology, 2004.

[15] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 439–449, 2004.

[16] A. Klapuri, "Multiple fundamental frequency estimation by summing harmonic amplitudes," in *in ISMIR*, pp. 216–221, 2006.

[17] A. Klapuri, "Sound Onset Detection by Applying Psychoacoustic Knowledge," in *Proceedings of the IEEE International Conference of Acoustics, Speech and Signal Processing*, vol. 6, (Washington, DC, USA), pp. 115–118, 1999.

[18] M. Helén and T. Virtanen, "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine," in *In: Proc. EUSIPCO 2005.*, 2005.

[19] T. Virtanen, "Separation of sound sources by convolutive sparse coding," in *in Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing, 2004. [Online] Available: http://journal.speech.cs.cmu.edu/SAPA2004*, 2004.

[20] D. Godsmark and G. J. Brown, "A blackboard architecture for computational auditory scene analysis,"

[21] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka, "Organization of hierarchical perceptual sounds – music scene analysis with autonomous processing modules and a quantitative information integration mechanism," in *Proc. of International Joint Conference on Artificial Intelligence (IJCAI)*, p. 158, Aug. 1995.

[22] M. Davy, S. Godsill, and J. Idier, "Bayesian analysis of polyphonic western tonal music.," *J Acoust Soc Am*, vol. 119, no. 4, pp. 2498–517, 2006.

[23] T. Virtanen, *Sound Source Separation in Monaural Music Signals*. PhD thesis, Tampere University of Technology, 2006.

[24] F. Hermansen, P. M. Bloomfield, J. Ashburner, P. G. Camici, and A. A. Lammertsma, "Linear dimension reduction of sequences of medical images: II. Direct sum decomposition," *Phys Med Biol*, vol. 40, pp. 1921–1941, Nov. 1995.

[25] J. Fine and A. Pouse, "Asympotic study of the multivariate functional model. application to the metric choice in principal component analysis," *Statistics: A Journal of Theoretical and Applied Statistics*, vol. 23, no. 1, pp. 63–83, 1992.

[26] V. Šmídl and A. Quinn, *The Variational Bayes Method in Signal Processing*. Springer, 2005.

[27] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, 2000.

[28] M. A. Casey and A. Westner, "Separation of Mixed Audio Sources by Independent Subspace Analysis," in *International Computer Music Conference*, pp. 154–161, 2000.

[29] D. Fitzgerald, E. Coyle, and B. Lawlor, "Sub-band independent subspace analysis for drum transcription," in *Proceedings of the 5th International Conference on Digital Audio Effects (DAFX'02)*, 2002.

[30] J.-F. Cardoso, "Multidimensional independent component analysis.," in *In Proc. Int. Workshop on Higher-Order Stat*, pp. 111–120, 1998.

[31] M. D. Plumbley and E. Oja, "A "nonnegative PCA" algorithm for independent component analysis," *Neural Networks, IEEE Transactions on*, vol. 15, no. 1, pp. 66–76, 2004.

[32] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *In NIPS*, pp. 556–562, MIT Press, 2001.

[33] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *NIPS*, pp. 556–562, 2000.

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

[35] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 13, no. 10, 2007.

[36] T. Virtanen and A. Klapuri, "Analysis of polyphonic audio using source-filter model and."

[37] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 177–180, 2003.

[38] J. Paulus and T. Virtanen, "Drum transcription with nonnegative spectrogram factorisation," in *IN EUSIPCO*, pp. 4–8, 2005.

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

[40] T. Virtanen and A. T. Cemgil, "Mixtures of gamma priors for non-negative matrix factorization based speech separation," in *ICA*, pp. 646–653, 2009.

[41] E. Vincent, N. Berlin, and R. Badeau, "Harmonic and inharmonic non-negative matrix factorization for polyphonic pitch transcription," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 109–112, 2008.

[42] R. Pérez, V. da Rocha Lopes, and e. C. d. C. Universidade Estadual de Campinas. Instituto de Matemática, Estatística, *Solving recent applications by quasi-Newton methods.* Relatório de pesquisa, Universidade Estadual de Campinas, 2001.

[43] S. A. Abdallah and M. D. Plumbley, "Unsupervised analysis of polyphonic music by sparse coding," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 179–196, 2006.

[44] E. Vincent and X. Rodet, "Music transcription with isa and hmm.," in *ICA* (C. G. Puntonet and A. Prieto, eds.), vol. 3195 of *Lecture Notes in Computer Science*, pp. 1197–1204, Springer, 2004.

[45] E. Vincent and M. Plumbley, D., "Fast factorization-based inference for Bayesian harmonic models," in *2006 IEEE Int. Workshop on Machine Learning for Signal Processing*, (Maynooth, Ireland), pp. 117–122, 2006.

[46] C. Dubois and M. Davy, "Joint detection and tracking of time-varying harmonic components: a flexible bayesian approach," in *IEEE transactions on Speech, Audio and Language Processing*, 2006.

[47] B. Raj and P. Smaragdis, "Latent variable decomposition of spectrograms for single channel speaker separation," in *in IEEE WASPAA*, pp. 17–20, 2005.

[48] P. Smaragdis, M. Shashanka, and B. Raj, "A sparse non-parametric approach for single channel separation of known sounds," 2009.

[49] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2 ed., Mar. 2008.

[50] P. S. Gautham J. Mysore, "Multipitch estimation using sparse impulse distributions and instrument specific priors," in *International Conference on Machine Learning (ICML)*, 2009.

[51] A. Klapuri, "A perceptually motivated multiple-f0 estimation method," in *in Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 291–294, 2005.

[52] T. Kitahara, Y. M. Goto, and H. G. O. Y, "Musical instrument identification based on f0-dependent multivariate normal distribution," in *in IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 421–424, 2003.

[53] G. Tzanetakis and G. Essl, "Automatic musical genre classification of audio signals," in *IEEE Transactions on Speech and Audio Processing*, pp. 293–302, 2001.

[54] I. Panagakis, E. Benetos, and C. Kotropoulos, "Music genre classification: A multilinear approach.," in *ISMIR* (J. P. Bello, E. Chew, and D. Turnbull, eds.), pp. 583–588, 2008.

[55] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *European Signal Processing Conference*, (Aalborg, North Denmark), 2010.

[56] K. Brandenburg, C. Dittmar, M. Gruhne, J. Abeßer, H. Lukashevich, P. Dunker, D. Gärtner, K. Wolter, and H. Grossmann, "Music search and recommendation," in *Handbook of Multimedia for Digital Entertainment and Arts* (B. Furht, ed.), pp. 349–384, Springer US, 2009. 10.1007/978-0-387-89024-1_16.

[57] E. D. Scheirer, Y. E. Kim, and E. Kim, "Generalized audio coding with mpeg-4 structured audio," 1999.

[58] I. O. for Standardization, I. E. Commission, and O. I. de Normalisation", *"ISO/IEC 14496-3:1999: Information Technology – coding of Audio-visual Objects – Part 3: Audio"*. International standard, ISO/IEC, 1999.

[59] M. Goto, "A predominant-f0 estimation method for real-world musical audio signals: Map estimation for incorporating prior knowledge about f0s and tone models," 2001.

[60] M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, (New Paltz, NY), 2005.

[61] R. Gribonval, L. Benaroya, E. Vincent, and C. Févotte, "Proposals for performance measurement in source separation," in *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, (Nara, Japan), pp. 763–768, 2003.

[62] Š. Albrecht, "Music signal decomposition approaches based on unsupervised source separation methods," in *8th International PhD Workshop on Systems and Control*, (Balatonfuhred, Hungary), Sept. 2007.

[63] Š. Albrecht, "Identification of components in music signal by sequential monte carlo," in *In Proc. International PhD Workshop on Systems and Control*, 2008.

[64] Š. Albrecht and V. Matoušek, "Music signal decomposition based on identification and subtraction of components," in *DAGA*, (Dresden, Germany), Mar. 2008.

[65] Š. Albrecht, "On the inverse music sequencer operation - detection of music components from wave table in complex music signal," in *DAGA*, (Rotterdam, Netherlands), 2009.

[66] Š. Albrecht and V. Šmídl, "Model considerations for memory-based automatic music transcription," in *29th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, (Oxford, Mississipi, US), 2009.

[67] Š. Albrecht, "Automatic music transcirption via music component identification," in *DAGA*, (Berlin, Germany), 2010.

[68] Š. Albrecht and V. Šmídl, "Improvements of continuous model for memory-based automatic music transcription," in *Proceedings of the 18th European signal processing conference, European signal processing conference*, (Aalborg, DK), 2010.

[69] Š. Albrecht and V. Šmídl, "Model considerations for memory-based automatic music transcription," in *Proceedings of the 19th European Signal Processing Conference*, (Barcelona, ES), 2011.

[70] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. Wiley, 1994.

[71] C. Robert, *The Bayesian choice: a decision-theoretic motivation*. Springer texts in statistics, Springer-Verlag, 1994.

[72] Z. Chen, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, pp. 1–69, 2003.

[73] M. S. Grewal and A. P. Andrews, *Kalman Filtering : Theory and Practice Using MATLAB*. Wiley-Interscience, 2 ed., Jan. 2001.

[74] V. Šmídl and A. Quinn, "Variational Bayesian filtering," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5020–5030, 2008.

[75] R. E. Kass and A. E. Raftery, "Bayes factors and model uncertainty," Tech. Rep. 571, Carnegie Mellon University Dept of Statistics, Pittsburgh, PA 15213, 1993.

[76] S. F. Gull, "Bayesian inductive inference and maximum entropy," *Maximum Entropy and Bayesian Methods in Science and Engineering*, pp. 53–74, 1988.

[77] W. J. F. Joseph J.K. O Ruanaidh, *The Bayesian choice: a decision-theoretic motivation*. Statistics and Computing, Springer, 1996.

[78] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics (Chapman & Hall/CRC Interdisciplinary Statistics)*. Chapman and Hall/CRC, 1 ed., Dec. 1995.

[79] C. Dubois and M. Davy, "Harmonic tracking using sequential monte carlo," in *IN SSP*, 2005.

[80] M. Davy and C. Dubois, "A fast particle filtering approach to bayesian tonal music transcription," 2007.

[81] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[82] S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–87, 1951.

[83] J. M. Bernardo, "Expected Information as Expected Utility," *Annals of Statistics*, vol. 7, pp. 686–690, 1979.

[84] T. S. Jaakkola and M. I. Jordan, "A variational approach to bayesian logistic regression models and their extensions," 1996.

[85] D. J. MacKay, "Developments in probabilistic modelling with neural networks - ensemble learning," 1995.

[86] M. J. Beal, *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

[87] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003. `http://www.inference.phy.cam.ac.uk/mackay/itprnn/book.html`.

[88] M. Sato, "Online model selection based on the variational Bayes," *Neural Computation*, vol. 13, pp. 1649–1681, 2001.

[89] C. Genest and J. V. Zidek, "Combining probability distributions: A critique and an annotated bibliography," *Statistical Science*, vol. 1, pp. 114–148, 1986.

[90] W. Briggs and V. Henson, *The DFT: an owner's manual for the discrete Fourier transform*. Miscellaneous Bks, Society for Industrial and Applied Mathematics, 1995.

[91] D. Schobben, K. Torkkola, and P. Smaragdis, "Evaluation of blind signal separation methods," pp. 261–266, 1999.

[92] http://www.opensound.com/proaudio-download.html.

[93] http://theremin.music.uiowa.edu/MIS.html.

[94] T. P. Minka, "Old and new matrix algebra useful for statistics," tech. rep., 2001.

[95] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. New York: Dover, ninth dover printing, tenth gpo printing ed., 1964.

Appendix A

Required Probability Distributions

Multivariate Normal distribution

The multivariate Normal distribution of $\boldsymbol{x} \in \mathbb{R}^{p+1}$ is

$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{R}) = (2\pi)^{-\frac{p}{2}} |\boldsymbol{R}|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2} [\boldsymbol{x} - \boldsymbol{\mu}]^T \boldsymbol{R}^{-1} [\boldsymbol{x} - \boldsymbol{\mu}] \right\}. \tag{1}$$

where $\boldsymbol{R}$ is symmetric, positive definite matrix. The non-zero moments of (1) are

$$\begin{aligned} \hat{\boldsymbol{x}} =& \boldsymbol{\mu}, \\ \widehat{\boldsymbol{x}\boldsymbol{x}^T} =& \boldsymbol{R} + \boldsymbol{\mu}\boldsymbol{\mu}^T. \end{aligned}$$

The scalar Normal distribution is a special case of (1):

$$\mathcal{N}(x, r) = (2\pi r)^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2r} (x - \mu)^2 \right\}. \tag{2}$$

Matrix Normal Distribution

The matrix Normal distribution of the matrix $\boldsymbol{X} \in \mathbb{R}^{p+n}$ is

$$\begin{aligned} \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{X}}, \boldsymbol{\Sigma}_p \otimes \boldsymbol{\Sigma}_n) =& (2\pi)^{-\frac{pn}{2}} |\boldsymbol{\Sigma}_p|^{-\frac{n}{2}} |\boldsymbol{\Sigma}_n|^{-\frac{p}{2}} \times \\ & \times \exp\left(-\frac{1}{2} \mathrm{tr}\left\{ \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{X} - \boldsymbol{\mu}_{\boldsymbol{X}}) (\boldsymbol{\Sigma}_n^{-1})^T (\boldsymbol{X} - \boldsymbol{\mu}_{\boldsymbol{X}})^T \right\} \right), \end{aligned}$$

where $\boldsymbol{\Sigma}_p \in \mathbb{R}^{p \times p}$ and $\boldsymbol{\Sigma}_n \in \mathbb{R}^{n \times n}$ are symmetric, positive definite matrices, and where $\otimes$ denotes the Kronecker product [94]. The distribution has the following properties:

- The first moment is $\mathsf{E}_{\boldsymbol{X}}[\boldsymbol{X}] = \boldsymbol{\mu}_{\boldsymbol{X}}$.

- The second non-central moments are

$$\begin{aligned}\mathsf{E}_{\boldsymbol{X}}[\boldsymbol{X}\boldsymbol{Z}\boldsymbol{X}^T] =& \mathrm{tr}(\boldsymbol{Z}\boldsymbol{\Sigma}_n)\boldsymbol{\Sigma}_p + \boldsymbol{\mu}_{\boldsymbol{X}}\boldsymbol{Z}\boldsymbol{\mu}_{\boldsymbol{X}}{}^T,\\ \mathsf{E}_{\boldsymbol{X}}[\boldsymbol{X}^T\boldsymbol{Z}\boldsymbol{X}] =& \mathrm{tr}(\boldsymbol{Z}\boldsymbol{\Sigma}_p)\boldsymbol{\Sigma}_n + \boldsymbol{\mu}_{\boldsymbol{X}}{}^T\boldsymbol{Z}\boldsymbol{\mu}_{\boldsymbol{X}},\end{aligned}$$

  where $\boldsymbol{Z}$ is an arbitrary matrix, appropriately resized in each case.

Gamma Distribution

Gamma distribution is as follows:

$$p(x|a,b) = \mathcal{G}(a,b) = \frac{b^a}{\Gamma(a)}x^{a-1}\exp(-bx)\chi_{[0,\infty)}(x), \tag{3}$$

where $a > 0$ and $b > 0$, $\chi_{[0,\infty)}(x)$ is the indicator function and $\Gamma(a)$ is the gamma function [95] evaluated at a. The first moment is

$$\hat{x} = \frac{a}{b}, \tag{4}$$

and the second central moment is

$$\mathsf{E}_{\boldsymbol{X}}[(x-\hat{x})^2] = \frac{a}{b^2}. \tag{5}$$

Multinomial Distribution

The Multinomial distribution of the c-dimensional vector variable l where $l_i \in \mathbb{N}$ and $\sum_{i=1}^{c} l_i = \gamma$ is as follows:

$$p(l|a) = \mathcal{M}u_l(\alpha) = \frac{1}{\zeta_l(\alpha)}\prod_{i=1}^{c}\alpha_i^{l_i}\chi_{\mathbb{N}^c}(l). \tag{6}$$

Its vector parameter is $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_c]^T$, $\alpha_i > 0$, $\sum_{i=1}^{c}\alpha_i = 1$. The indicator function [95] is denoted by $\chi_{\mathbb{N}^c}(l)$. The normalizing constant is

$$\zeta_l(\alpha) = \frac{\prod_{i=1}^{c} l_i!}{\gamma!}, \tag{7}$$

where "!" denotes factorial.

If the argument l contains positive real numbers, i.e., $l_i \in (0,\infty)$, then we refer to (6) as the Multinomial distribution of continuous argument. The only change in (6) is that the support is now $(0,\infty)^c$ and the normalizing constant is

$$\zeta_l(\alpha) = \frac{\prod_{i=1}^{c}\Gamma(l_i)}{\Gamma(\gamma)}, \tag{8}$$

where $\Gamma(\cdot)$ is the gamma function [95]. For both variants the first moment is given by

$$\hat{l} = \alpha. \tag{9}$$

Author's Publications

Contributions to Conferences Cited by Conference Proceedings Citation Index (ISI)

[1] Š. Albrecht and V. Šmídl, "Model considerations for memory-based automatic music transcription," in *Proceedings of the 19th European Signal Processing Conference*, (Barcelona, ES), 2011.

[2] Š. Albrecht and V. Šmídl, "Improvements of continuous model for memory-based automatic music transcription," in *Proceedings of the 18th European signal processing conference, European signal processing conference*, (Aalborg, DK), 2010.

[3] Š. Albrecht and V. Šmídl, "Improvements of continuous model for memory-based automatic music transcription," in *29th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, (Oxford, Mississipi, US), 2009.

Contributions to Other Conferences

[1] Š. Albrecht, "Automatic music transcirption via music component identification," in *DAGA*, (Berlin, Germany), 2010.

[2] Š. Albrecht, "On the inverse music sequencer operation - detection of music components from wave table in complex music signal," in *DAGA*, (Rotterdam, Netherlands), 2009.

[3] Š. Albrecht, "Identification of components in music signal by sequential monte carlo," in *In Proc. International PhD Workshop on Systems and Control*, 2008.

[4] Š. Albrecht and V. Matoušek, "Music signal decomposition based on identification and subtraction of components," in *DAGA*, (Dresden, Germany), Mar. 2008.

[5] Š. Albrecht, "Music signal decomposition approaches based on unsupervised source separation methods," in *8th International PhD Workshop on Systems and Control*, (Balatonfuhred, Hungary), Sept. 2007.