Queen Mary University of London

centre for digital music

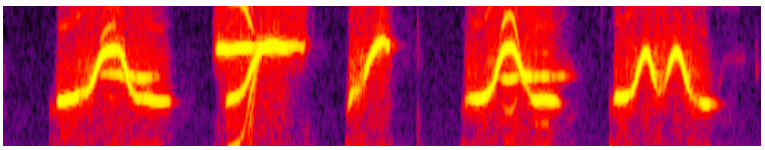
Master's Thesis

High Quality Musical Audio Source Separation

Host institution : Centre for Digital Music - QMUL

Supervisor : Mark D. Plumbley

Master ATIAM
(UPMC - IRCAM - Telecom ParisTech)



Joachim Fritsch - August 2012

UPMC SORBONNE UNIVERSITÉS

ircam Centre Pompidou

TELECOM ParisTech

Summary

This Master's thesis is dedicated to the challenging problem of high quality separation of musical audio sources in monophonic mixtures, using information from musical scores to help improve the quality of separation. This information is integrated in a Nonnegative Matrix Factorization (NMF) of the audio spectrogram, providing essential temporal and spectral information to guide the decomposition process. A general framework is proposed for this decomposition process, where the activation coefficients and the basis functions of each instruments are initially learnt on synthesized signals, and then used to initialize the decomposition of the actual mixture. This method is applied to an existing dataset and assessed with the BSS_EVAL and PEASS evaluation toolboxes. The performance measures are then compared with those obtained with another method from the literature, and a new dataset is created in order to study the influence of the various parameters on the separation results.

Ce mémoire de Master est dédié au stimulant problème de la séparation de sources musicales de haute qualité dans des enregistrements monophoniques, en utilisant les informations fournies par la partition musicale pour aider à améliorer la qualité de séparation. Ces informations sont intégrées dans une Factorisation en Matrices Non-Négatives (NMF en anglais) du spectrogramme audio, fournissant ainsi des informations temporelles et fréquentielles essentielles pour guider le processus de décomposition. Un cadre général est proposé pour ce processus de décomposition, dans lequel les coefficients d'activation et les fonctions de base de chaque instrument sont initialement appris sur des signaux de synthèse, puis utilisés pour initialiser la décomposition de l'enregistrement en question. Cette méthode est appliquée sur un set de données pré-existant, et évaluée à l'aide des toolboxes BSS_EVAL et PEASS. Les mesures de performances sont alors comparées à celles obtenues avec une autre méthode issue de la littérature, et un nouveau set de données est créé afin d'étudier l'influence des différents paramètres sur les résultats de séparation.

Acknowledgements

I would like to thank my tutor Mark Plumbley, who suggested the project to me and who supervised my work with a constant enthusiasm and a lot of generosity.

I would also like to thank Laurent Daudet, for his precious support during my placement application and for his great availability.

From the Centre for Digital Music, I would like to thank Daniele and Dimitrios for their kind welcome, Maria for presenting her research to me, Emmanouil for his help with Sonic Visualizer, Dan and Ken for their enjoyable company in Edinburgh, Steven for his inimitable British humour, and all the other people who made my visit at the Centre very pleasant.

I am also very grateful to Joachim Ganseman for his valuable help while writing my first research paper, and I would like to give a special thank to my desk-mate Holger Kirchhoff, for his incredible support and for contributing so much to my work through daily discussions.

My thanks also go to Romain Hennequin, for his explanations on his score-informed source separation method, and to Cédric Févotte, for his notes and comments on my work and for his accurate details on the Smooth Itakura-Saito NMF.

I would also like to thank Thibault, Divyang and all the people from Lee Abbey International Student Club, for their friendship and their kind encouragement.

Finally, I would like to thank Emma and my parents for their love, their patience and their multiple efforts to understand my research topic!

Table of Contents

Introduction

The purpose of source separation applied to musical audio is to separate the signal of each instrument in a polyphonic mixture. This separation is generally achieved through the decomposition of a time-frequency representation of the mixture signal such as the Short Time Fourier Transform (STFT), commonly called spectrogram, or the Constant-Q Transform (CQT).

This decomposition is allowed by a series of matrix factorization techniques developed in the last decade, among which Nonnegative Matrix Factorization (NMF) and Independent Component Analysis (ICA) appeared to have a great success.

Once separated, the different musical sources can be used individually, or processed separately and reassembled eventually. Musical audio source separation can therefore have potential applications such as music remastering, audio denoising, upmixing (mono to stereo), etc.

In our personal approach, we attempt to improve the quality of separation by using information provided from a symbolic representation of the musical signal, *i.e.* an aligned version of the written score. This approach has already been addressed in the literature under the name of "score-informed source separation", and various methods have therefore been elaborated in the past few years.

In this present work, we collect some ideas and techniques from these different methods in order to propose our own framework for score-informed source separation. We initially present the NMF algorithm used in our decomposition process, and then we describe the different steps of this latter in a second part. We present afterwards the various experiments run to assess the quality of our proposed method, and finally we discuss the different results.

1 Musical Audio Source Separation using NMF

1.1 Nonnegative Matrix Factorization

1.1.1 Principle and standard problem

Nonnegative Matrix Factorization (NMF) is a low-rank approximation technique, used for part-based decomposition of nonnegative data. Given a matrix $\mathbf{V}$ of dimensions $F \times N$ with nonnegative entries, NMF is the problem of finding a factorization

$$\mathbf{V} \approx \mathbf{WH} \tag{1}$$

where $\mathbf{W}$ and $\mathbf{H}$ are nonnegative matrices of dimensions $F \times K$ and $K \times N$ respectively. K represents the rank (*i.e.* the number of components) of the decomposition, and is usually chosen such that $FK + KN \ll FN$, hence reducing the data dimension [1].

NMF has been originally presented in [2] and [3], and has been applied afterwards to diverse problems (such as pattern recognition, clustering, data mining) in various areas (such as bioinformatics, signal and image processing, finance).

The matrix $\mathbf{W}$ is usually referred to as the "dictionary", with each column representing the basis functions of the different components, and the rows of $\mathbf{H}$ represent the "activation coefficients" of these components over the other dimension. Each column $\mathbf{v}_n$ of the original data is therefore approximated by a linear combination of the basis functions of $\mathbf{W}$, with the corresponding activation coefficients $\mathbf{h}_n$ [1].

$$\mathbf{v}_n \approx \mathbf{W}\mathbf{h}_n \tag{2}$$

A distinctive characteristic of NMF, in comparison with other factorization techniques such as Vector Quantization (VQ), Principal Component Analysis (PCA) or Independent Component Analysis (ICA), is the nonnegativity constraint on the factorized matrices $\mathbf{W}$ and $\mathbf{H}$, which improves the interpretability of the learnt dictionary and the activation coefficients when the original data is nonnegative. Indeed, the nonnegativity of $\mathbf{W}$ allows the learnt basis functions to belong to the same space than the data, and the nonnegativity of $\mathbf{H}$ ensures a constructive representation of the data, as subtractive combinations are forbidden [1].

The factorization (1) is usually sought after through the minimization problem

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{V}|\mathbf{WH}) \text{ subject to } \mathbf{W} \geq 0, \mathbf{H} \geq 0 \tag{3}$$

where the notation $\mathbf{A} \geq 0$ expresses the nonnegativity restriction on the entries of matrix $\mathbf{A}$, and where $D(\mathbf{V}|\mathbf{WH})$ is generally a separable measure of fit such that

$$D(\mathbf{V}|\mathbf{WH}) = \sum_{f=1}^{F}\sum_{n=1}^{N} d([\mathbf{V}]_{f,n}|[\mathbf{WH}]_{f,n}) \tag{4}$$

where $d(x|y)$ is a scalar cost function, typically a positive function of $y \in \mathbb{R}_+$ given $x \in \mathbb{R}_+$, with a single minimum for $x = y$ [1].

1.1.2 Use of the β-divergence as a cost function

A commonly used cost function in NMF is the β-divergence $d_\beta(x|y)$, originally introduced in [4] and [5] and defined rigorously in [6] as following

$$d_\beta(x|y) \stackrel{\text{def}}{=} \begin{cases} \frac{1}{\beta(\beta-1)}(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}) & \beta \in \mathbb{R}\backslash\{0,1\} \\ x\log\frac{x}{y} - x + y & \beta = 1 \\ \frac{x}{y} - \log\frac{x}{y} - 1 & \beta = 0. \end{cases} \tag{5}$$

The β-divergence is thus defined by its single parameter β, and takes the Euclidean distance, the generalized Kullback-Leibler (KL) divergence and the Itakura-Saito (IS) divergence as special cases ($\beta = 2$, 1 and 0, respectively).

If we introduce the approximation of the data $\hat{\mathbf{V}}$ such that

$$\hat{\mathbf{V}} = \mathbf{WH} = \sum_{k=1}^{K} \mathbf{W}_{f,k}\mathbf{H}_{k,n} \tag{6}$$

where the notation $\mathbf{A}_{i,j}$ represents the entry (i,j) of the matrix $\mathbf{A}$ (usually denoted as $[\mathbf{A}]_{i,j}$), then NMF with the β-divergence (henceforth shortened as "β-NMF") is the minimization problem (3) with the following cost function

$$C_\beta = D_\beta(\mathbf{V}|\hat{\mathbf{V}}) = \sum_{f=1}^{F}\sum_{n=1}^{N} \frac{1}{\beta(\beta-1)}\left(\mathbf{V}_{f,n}^\beta + (\beta-1)\hat{\mathbf{V}}_{f,n}^\beta - \beta\mathbf{V}_{f,n}\hat{\mathbf{V}}_{f,n}^{\beta-1}\right) \tag{7}$$

for $\beta \in \mathbb{R}\backslash\{0,1\}$ and

$$C_{KL} = \sum_{f=1}^{F}\sum_{n=1}^{N} \mathbf{V}_{f,n}\log\frac{\mathbf{V}_{f,n}}{\hat{\mathbf{V}}_{f,n}} - \mathbf{V}_{f,n} + \hat{\mathbf{V}}_{f,n} \tag{8}$$

$$C_{IS} = \sum_{f=1}^{F}\sum_{n=1}^{N} \frac{\mathbf{V}_{f,n}}{\hat{\mathbf{V}}_{f,n}} - \log\frac{\mathbf{V}_{f,n}}{\hat{\mathbf{V}}_{f,n}} - 1 \tag{9}$$

for the Kullback-Leibler and Itakura-Saito special cases.

1.1.3 Resolution with a gradient descent algorithm

A very popular optimization strategy in NMF is based on the iteration of multiplicative updates, where the matrices $\mathbf{W}$ and $\mathbf{H}$ are optimized alternatively through a gradient descent. If we call θ the components of these matrices and η the step size, the gradient descent update rule with the β-divergence is given by

$$\theta \leftarrow \tilde{\theta} - \eta\frac{\partial C_\beta}{\partial\tilde{\theta}} \tag{10}$$

where $\tilde{\theta}$ denotes the old component θ, before the update.

The multiplicative updates mentioned above are then obtained by setting the step size η analytically, so that the update rule (10) becomes multiplicative. We will now show how to obtain these multiplicative update rules for $\mathbf{W}$ and $\mathbf{H}$, with a demonstration based on the technical report [7] used in [8].

Update rule for W

If we denote $\mathbf{W}_{\tilde{f},\tilde{k}}$ a single element of the matrix $\mathbf{W}$, the gradient descent update rule with the β-divergence for this single element is given by

$$\mathbf{W}_{\tilde{f},\tilde{k}} \leftarrow \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}} - \eta \frac{\partial C_\beta}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}}. \tag{11}$$

We first derive an auxiliary term which will appear in the differentiation of the cost function C_β, and which is the differentiation of $\hat{\mathbf{V}}$ w.r.t. the coefficient $\mathbf{W}_{\tilde{f},\tilde{k}}$.

$$\begin{aligned}
\frac{\partial \hat{\mathbf{V}}}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}} &= \frac{\partial}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}} \left(\sum_{k=1}^{K} \mathbf{W}_{f,k} \mathbf{H}_{k,n} \right) \\
&= \frac{\partial}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}} \left(\mathbf{W}_{f,\tilde{k}} \mathbf{H}_{\tilde{k},n} \right) \\
&= \begin{cases} \mathbf{H}_{\tilde{k},n} & \text{if } f = \tilde{f} \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{12}$$

We can then derive the gradient of the cost function C_β w.r.t. the single element $\mathbf{W}_{\tilde{f},\tilde{k}}$. The sum over f in the expression of the cost function C_β can be dropped in the differentiation, as all the terms under this sum are regarded as constant and vanish during the derivation, expect the one containing $\tilde{f}$.

$$\begin{aligned}
\frac{\partial C_\beta}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}} &= \frac{\partial C_\beta}{\partial \hat{\mathbf{V}}} \cdot \frac{\partial \hat{\mathbf{V}}}{\partial \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}} \\
&= \sum_{n=1}^{N} \frac{1}{\beta(\beta-1)} \left(\beta(\beta-1) \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-1} - \beta(\beta-1) \mathbf{V}_{\tilde{f},n} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-2} \right) \mathbf{H}_{\tilde{k},n} \\
&= \sum_{n=1}^{N} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-1} \mathbf{H}_{\tilde{k},n} - \mathbf{V}_{\tilde{f},n} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-2} \mathbf{H}_{\tilde{k},n}
\end{aligned} \tag{13}$$

Given this result, the gradient descent update rule (11) thus becomes

$$\mathbf{W}_{\tilde{f},\tilde{k}} \leftarrow \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}} - \eta \left(\sum_{n=1}^{N} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-1} \mathbf{H}_{\tilde{k},n} - \mathbf{V}_{\tilde{f},n} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-2} \mathbf{H}_{\tilde{k},n} \right). \tag{14}$$

If we set the step size η as following

$$\eta = \frac{\tilde{\mathbf{W}}_{\tilde{f},\tilde{k}}}{\sum_{n=1}^{N} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-1} \mathbf{H}_{\tilde{k},n}}, \tag{15}$$

the gradient descent update rule (14) is therefore simplified, and we obtain the desired multiplicative update form.

$$\mathbf{W}_{\tilde{f},\tilde{k}} \leftarrow \tilde{\mathbf{W}}_{\tilde{f},\tilde{k}} \cdot \frac{\sum_{n=1}^{N} \mathbf{V}_{\tilde{f},n} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-2} \mathbf{H}_{\tilde{k},n}}{\sum_{n=1}^{N} \hat{\mathbf{V}}_{\tilde{f},n}^{\beta-1} \mathbf{H}_{\tilde{k},n}} \tag{16}$$

Finally, we can express this multiplicative update in a very convenient matrix form

$$\mathbf{W} \leftarrow \mathbf{W} \bullet \frac{\left(\mathbf{V} \bullet \hat{\mathbf{V}}^{\bullet[\beta-2]}\right)\mathbf{H}^T}{\left(\hat{\mathbf{V}}^{\bullet[\beta-1]}\right)\mathbf{H}^T}, \tag{17}$$

where the symbol $\bullet$ denotes element-wise operations (multiplication, division and exponentiation), and where all the elements of $\mathbf{W}$ are updated simultaneously.

Update rule for H

The multiplicative update rule for the matrix $\mathbf{H}$ is obtained in a similar way to that for the matrix $\mathbf{W}$. If we denote $\mathbf{H}_{\tilde{k},\tilde{n}}$ a single element of the matrix $\mathbf{H}_{k,n}$, the gradient descent update rule with the β-divergence is given in this case by

$$\mathbf{H}_{\tilde{k},\tilde{n}} \leftarrow \tilde{\mathbf{H}}_{\tilde{k},\tilde{n}} - \eta \frac{\partial C_\beta}{\partial \tilde{\mathbf{H}}_{\tilde{k},\tilde{n}}}. \tag{18}$$

The differentiation of the cost function C_β w.r.t. the element $\mathbf{H}_{\tilde{k},\tilde{n}}$ is obtained as in the equations (12) and (13), with an equivalent auxiliary term.

$$\frac{\partial C_\beta}{\partial \tilde{\mathbf{H}}_{\tilde{k},\tilde{n}}} = \sum_{f=1}^{F} \hat{\mathbf{V}}_{f,\tilde{n}}^{\beta-1}\mathbf{W}_{f,\tilde{k}} - \mathbf{V}_{f,\tilde{n}}\hat{\mathbf{V}}_{f,\tilde{n}}^{\beta-2}\mathbf{W}_{f,\tilde{k}} \tag{19}$$

If we set the step size η such that

$$\eta = \frac{\tilde{\mathbf{H}}_{\tilde{k},\tilde{n}}}{\sum_{f=1}^{F} \hat{\mathbf{V}}_{f,\tilde{n}}^{\beta-1}\mathbf{W}_{f,\tilde{k}}}, \tag{20}$$

we can simplify again the gradient descent update rule (18) in order to obtain the desired multiplicative update form

$$\mathbf{H}_{\tilde{k},\tilde{n}} \leftarrow \tilde{\mathbf{H}}_{\tilde{k},\tilde{n}} \cdot \frac{\sum_{f=1}^{F} \mathbf{V}_{f,\tilde{n}}\hat{\mathbf{V}}_{f,\tilde{n}}^{\beta-2}\mathbf{W}_{f,\tilde{k}}}{\sum_{f=1}^{F} \hat{\mathbf{V}}_{f,\tilde{n}}^{\beta-1}\mathbf{W}_{f,\tilde{k}}}, \tag{21}$$

which can also be expressed in the following matrix form

$$\mathbf{H} \leftarrow \mathbf{H} \bullet \frac{\mathbf{W}^T\left(\mathbf{V} \bullet \hat{\mathbf{V}}^{\bullet[\beta-2]}\right)}{\mathbf{W}^T\left(\hat{\mathbf{V}}^{\bullet[\beta-1]}\right)}. \tag{22}$$

The great advantage of the multiplicative update rules (17) and (22) is to ensure the nonnegativity restriction on the matrices $\mathbf{W}$ and $\mathbf{H}$, as we only multiply nonnegative terms during the updates. They are also very easy to implement and usually give simple and fast algorithms. The disadvantages are the impossibility to set the step size manually, and also the potential divisions by 0 during the updates, that can be avoided in practice by adding a small constant to the denominators.

Presentation of the algorithm

We can summarize the gradient descent algorithm with multiplicative updates for β-NMF with the following pseudocode (see Algorithm 1), as presented in [1].

Algorithm 1 β-NMF with multiplicative updates

Input: nonnegative matrix $\mathbf{V}$
Output: nonnegative matrices $\mathbf{W}$ and $\mathbf{H}$ such that $\mathbf{V} \approx \mathbf{WH}$
  Initialize $\mathbf{W}$ and $\mathbf{H}$ with nonnegatives values
  for $i = 1 : n_{iter}$ **do**
    Compute $\hat{\mathbf{V}} = \mathbf{WH}$
    $\mathbf{W} \leftarrow \mathbf{W} \bullet \dfrac{(\mathbf{V} \bullet \hat{\mathbf{V}}^{\bullet[\beta-2]})\mathbf{H}^T}{(\hat{\mathbf{V}}^{\bullet[\beta-1]})\mathbf{H}^T}$
    Compute $\hat{\mathbf{V}} = \mathbf{WH}$
    $\mathbf{H} \leftarrow \mathbf{H} \bullet \dfrac{\mathbf{W}^T(\mathbf{V} \bullet \hat{\mathbf{V}}^{\bullet[\beta-2]})}{\mathbf{W}^T(\hat{\mathbf{V}}^{\bullet[\beta-1]})}$
    Normalize $\mathbf{W}$ and $\mathbf{H}$
  end for

In this algorithm, the initialization of $\mathbf{W}$ and $\mathbf{H}$ is not specified. We will see in Section 2.2 that it has a great impact on the decomposition process, as it allows to provide prior information and to incorporate constraints on both matrices.

This algorithm also includes a normalization step at every iteration, which eliminates trivial scale indeterminacies leaving the cost function unchanged. This normalization step can be important when we intend to compare the amplitudes of the basis functions or the activation coefficients after the factorization process. In practice, we impose $||\mathbf{W}_k||_1 = 1$ and scale $\mathbf{H}_k$ accordingly [1].

1.2 Application to Musical Audio Source Separation

1.2.1 Factorization of the audio spectrogram

When NMF is applied to audio, the nonnegative data $\mathbf{V}$ is usually taken as the magnitude (or power) spectrogram of the signal, while the basis functions of the dictionary $\mathbf{W}$ are magnitude (or power) spectra, being activated over time according to the amplitudes contained in $\mathbf{H}$. This decomposition is indeed well suited for the composite structure of audio signals, as it represents constructive combinations of spectral features (or "sound objects") over time.

The first audio application of NMF has been automatic music transcription [9], followed by musical audio source separation [10]. In these musical cases, the NMF model takes anew advantage of the redundant characteristic of music, as this latter can be defined in a restrictive way as a limited number of notes or instruments being played over time.

An example of this musical decomposition is given in Figure 1, where the magnitude spectrogram of two piano notes $\mathbf{V}$ is approximated by a dictionary $\mathbf{W}$ of two magnitude spectra, multiplied by their corresponding amplitudes in $\mathbf{H}$. The two notes are first being played separately, and then played together.

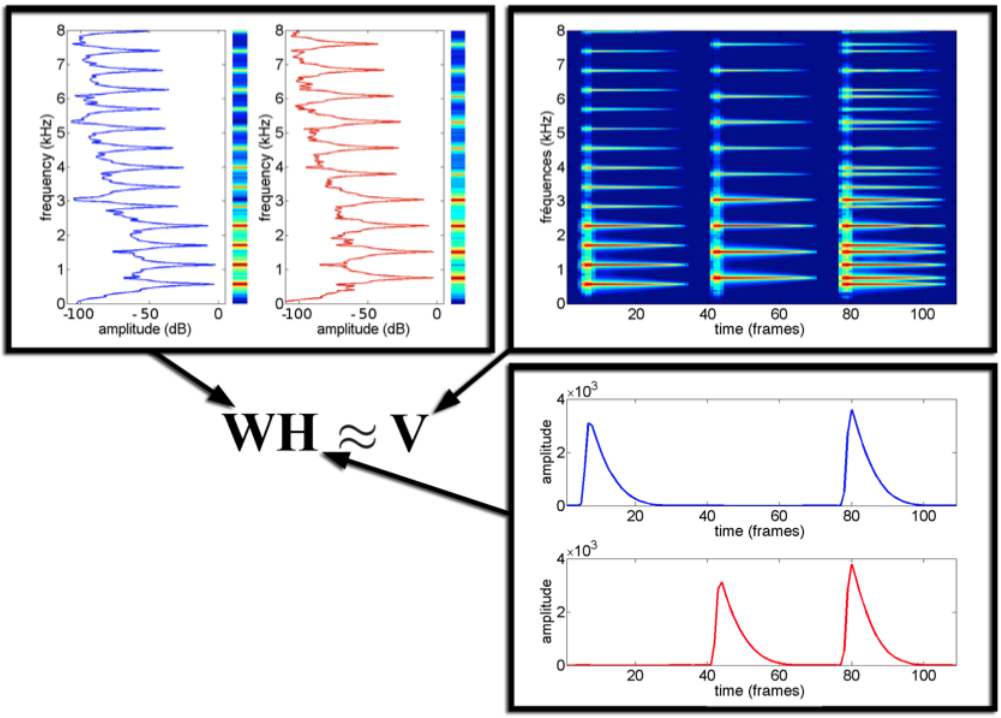


Figure 1: NMF applied to the audio spectrogram [11]

1.2.2 Component reconstruction and source extraction

Once the decomposition of the audio spectrogram is over, we obtain the estimated complex spectrogram $\hat{\mathbf{X}}^{(\tilde{c})}$ of a single component $\tilde{c}$ by using a Wiener filtering technique. This consists of applying a time-frequency mask (created by multiplying the basis function $\mathbf{W}_{f,\tilde{c}}$ and the amplitude $\mathbf{H}_{\tilde{c},n}$ of the component $\tilde{c}$, and by normalizing the resulting matrix by $\hat{\mathbf{V}}$) on the original complex spectrogram $\mathbf{X}$.

$$\hat{\mathbf{X}}^{(\tilde{c})} = \frac{\mathbf{W}_{f,\tilde{c}}\mathbf{H}_{\tilde{c},n}}{\sum_{k=1}^{K} \mathbf{W}_{f,k}\mathbf{H}_{k,n}} \bullet \mathbf{X} \tag{23}$$

This extraction method allows phase reconstruction, and also ensures a conservative decomposition of the original signal, as demonstrated in [1].

$$\mathbf{X} = \sum_{c=1}^{K} \hat{\mathbf{X}}^{(c)} \tag{24}$$

Eventually, we extract the complex spectrogram $\hat{\mathbf{X}}^{(s)}$ of a musical source compound of C components by simply adding the time-frequency masks of each component, as the Short-Time Fourier Transform (STFT) is a linear operation. The time signal $\hat{\mathbf{x}}^{(s)}$ of the extracted source is thus obtained through the inverse STFT of $\hat{\mathbf{X}}^{(s)}$.

$$\hat{\mathbf{X}}^{(s)} = \frac{\sum_{c=1}^{C} \mathbf{W}_{f,c}\mathbf{H}_{c,n}}{\sum_{k=1}^{K} \mathbf{W}_{f,k}\mathbf{H}_{k,n}} \bullet \mathbf{X} \xrightarrow{\text{STFT}^{-1}} \hat{\mathbf{x}}^{(s)} \tag{25}$$

2 Score-Informed Source Separation

2.1 Introduction

A musical score provides a wide range of information, such as the pitch, the onset time and the duration of each note played by each instrument. This information can therefore be used to provide temporal and spectral indications on the musical signal we seek to decompose, such as the frequencies of the harmonics or the temporal envelopes of the different sources present in the mixture.

In the past few years, many attempts have been made to supervise the source separation process in such a way, henceforth known as score-informed source separation. In [12], the knowledge of the written score is used with spatial cues to accurately separate time-frequency bins in a stereophonic mixture. In [13], the score of the solo part helps to separate it from the accompaniment with a classifier approach. In [14], the score is substituted by a "humming" query, thus used as prior in a Probabilistic Latent Component Analysis (PLCA) decomposition of the mixture. This approach is extended in [15], where the "humming" query is replaced by artificial signals synthesized from the score. In [16], the information from the score is used to initialize an algorithm based on a parametric decomposition of the spectrogram, using an original NMF framework. In [17] finally, the separation is performed in real-time, with a score-follower using a hidden Markov approach and a source separator extracting the different harmonics of each instrument.

In our personal approach, we use the NMF algorithm presented in Section 1.1.3 with temporal and harmonic constraints inspired from [16], and we add afterwards a preliminary learning phase on synthesized signals, as performed in [15]. We decide to work on monophonic signals only, thus ignoring the information provided from spatial cues. We do not consider the problem of score-to-audio alignment, and so we only use scores already aligned in a MIDI format.

2.2 Description of the original method

As mentioned in Section 1.1.3, the information provided from the score can be integrated in the NMF decomposition process through the initialization of the matrices $\mathbf{H}$ and $\mathbf{W}$. These matrices are therefore not initialized randomly, as it is done in what we call Blind Source Separation (BSS), but with specific initializations.

In our original method we only consider harmonic instruments, and so we assign one component per note per instrument in the factorized representation of the mixture. In order to collect the residuals sounds, we also add some extra-component with random initializations, as we will see in Section 2.2.2.

2.2.1 Temporal constraint on H

After extracting the onset and offset times from the aligned MIDI file of the score, we initialize the activation coefficients of each note by a simple binary function, equal to 1 if the note is being played and equal to 0 if not (we ignore the information provided by the MIDI velocity). This creates a "pianoroll" representation of the score, which is then used for the initialization of the matrix $\mathbf{H}$ (see Figure 2).

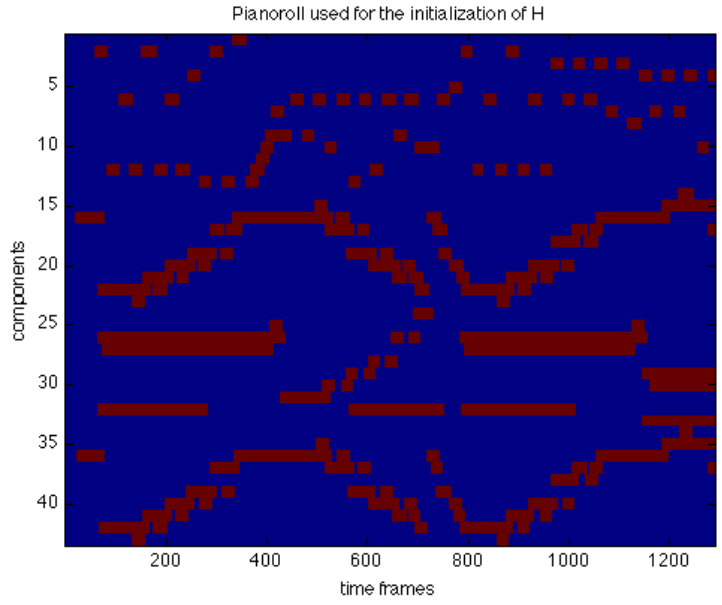


Figure 2: Pianoroll used for the initialization of **H**

The advantage of this binary initialization is to incorporate a temporal constraint on the elements of **H**, as the coefficients initialized to 0 will remain to 0 over the iterations, owing to the multiplicative updates used in the NMF algorithm. The moments of silence will therefore remain silent, and only the coefficients initialized to 1 will fit the actual temporal envelopes of the corresponding notes.

This constraint helps the algorithm to learn only the appropriate notes in a single time frame, and also present the great advantage to associate each component with a specific source. The separation process is consequently much improved, as we do not need to identify the extracted components as in BSS.

In practice, we usually enlarge slightly the initializations to 1 at the beginning and the end of each note, to avoid possible alignment errors and to take the possibly slow release of a note into account.

2.2.2 Harmonic constraint on W

The temporal constraint on **H** allows the algorithm to dissociate the learning of the different notes along time, but if many notes are being played during the same time frames, there is no guarantee that the decomposition will lead to the factorization of one note per component.

To solve this problem, we introduce a harmonic constraint on the basis functions of the dictionary, by initializing the spectra of each components with a harmonic comb adapted to the fundamental frequency of the corresponding notes. This creates a collection of harmonic combs, then used to initialize the matrix **W** (see Figure 3). In this model, the number of harmonics is set manually for each component, and we usually chose a fixed number for each instrument.

This harmonic constraint helps the algorithm to segregate the notes being played simultaneously by differentiating them according to their pitch (*i.e.* their harmonic structure), and thus improves the factorization results.

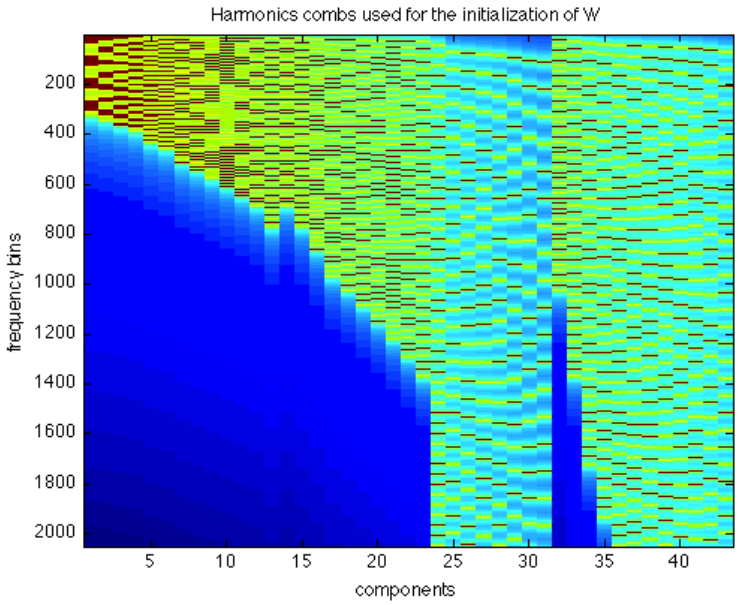


Figure 3: Harmonic combs used for the initialization of **W**

We build the so-called harmonic combs by convolving the magnitude (or power) spectrum of the analysis window by a Dirac comb adapted to the fundamental frequency of the corresponding notes [16]. This is done in order to adapt the peaks of the harmonic combs to the frequency resolution of the spectrogram, as we can quickly demonstrate.

If we call $x(t)$ an impulse train of infinite duration (whose spectrum $X(f)$ is thus a perfect Dirac comb), and $w(t)$ the analysis window of the STFT, the spectrum $Y(f)$ of the observed signal $y(t)$ is then given by

$$\begin{aligned} x(t) \cdot w(t) &= y(t) \\ (X * W)(f) &= Y(f). \end{aligned} \tag{26}$$

With an infinite frequency resolution, the harmonic model for the basis functions of **W** would be the perfect Dirac comb $X(f)$, but as the harmonic signals are windowed for each time frame of the STFT, we convolve that Dirac comb by the spectrum $W(f)$ of the analysis window. The type and the length of the analysis window are therefore significant and determine the width of the peaks and the dynamic of the harmonic comb used as a model.

As mentioned above, we add some extra-components with random activations to collect the residuals sounds during the decomposition process. Indeed, the harmonic model that we have just presented is not suited for the noise part of the musical signals that we intend to factorize, such as the impacts, blowing, clapping, plucking, or any other instrumental sounds.

In practice, we simply initialize the basis functions of these extra-components by a uniform distribution scaled between 0 and the maximum of the harmonic comb $Y(f)$, and the corresponding activation coefficients by a uniform distribution scaled between 0 and 1, in order to have the same scales for all the components.

This method creates a harmonic/noise separation of the original signal by grouping all the residual sounds in an additional "extra-source", which is not desirable but is an acceptable compromise when we want to improve the factorization results with the use of a harmonic constraint[1].

2.3 Description of the improved method

The temporal and harmonic constraints on **H** and **W** help the NMF algorithm to obtain the factorization of one note per component, but another problem arrises when two or more instruments are playing in unison (or in a harmonic interval such as an octave or a fifth). In such cases, the initialization of the affected components is indeed identical (or very similar), and once again there is no guarantee that the algorithm will factorize the contribution of each instrument in the corresponding components separately.

To solve this problem, we need to initialize the amplitudes and the spectra of each instruments in a distinctive way, corresponding to the physics of the instruments. This is done in our improved method by learning these amplitudes and spectra on a synthesized version of the score, with a method also known as "score synthesis".

2.3.1 Score synthesis and preliminary learning phase

The idea of score synthesis is to use a synthesized version of the instrumental signals generated from the score as a model for the factorization algorithm. The components of the different instruments are for this estimated from these separated signals in a preliminary learning phase, and then used to initialize the decomposition of the actual musical mixture.

This idea has been introduced in [15] with the use of a PLCA factorization technique, very similar to the NMF approach. In their method, the authors used a certain number of components with random initializations to learn the activation coefficients and the basis functions of each instrument separately, and then joined them together to initialize the "unmixing" phase.

In our personal score synthesis method, we decide to keep the temporal and harmonic constraints presented in Section 2.2 during the learning phase, in order to preserve the factorization form of one component per note per instrument. The contribution of the preliminary learning phase is therefore to provide distinctive models for the temporal and spectral envelopes of each instrument, with the advantage mentioned above for the separation of instruments playing in unison or in harmonic intervals.

These models could be supplied to the NMF algorithm otherwise, with the use of analytical models for the physics of each instruments for example, but the advantage of score synthesis is to be fast and easy to implement and to be adaptable for every type of instrument. On the other side, the reliability of the learnt data depends on the quality of the synthesizer. In our case we use thus a sample-based synthesizer, supposed to provide signals very close to the physical reality.

[1] A further discussion about this problem can be found in the conclusion of this thesis.

2.3.2 Presentation of the general framework

The general framework of our improved method is presented in Figure 4.

In the preliminary learning phase, the components of each instruments are learnt separately on the spectrogram of the synthesized signals $\mathbf{V_{syn}}$. The NMF routines are initialized with the collections of harmonic combs $\mathbf{W_0}$ and the pianorolls $\mathbf{H_0}$ generated from the score parts, and the residual sounds are collected with the addition of some extra-components. These extra-components are thereafter set apart and ignored, as the residuals from the synthesized signals are not likely to have the same spectral structure than the residuals from the actual signals.

In the unmixing phase, the basis functions and the activation coefficients learnt from previous phase are grouped into the matrices $\mathbf{W_1}$ and $\mathbf{H_1}$, then used to initialize another NMF routine on the spectrogram of the actual mixture $\mathbf{V_{mix}}$. The residuals sounds are again collected with some extra-components, and constitute an additional "extra-source" after the decomposition, as explained above.

At last, the different instruments are extracted from the matrices $\mathbf{W}$ and $\mathbf{H}$ resulting from the unmixing phase, with the Wiener filtering technique presented in Section 1.2.2.

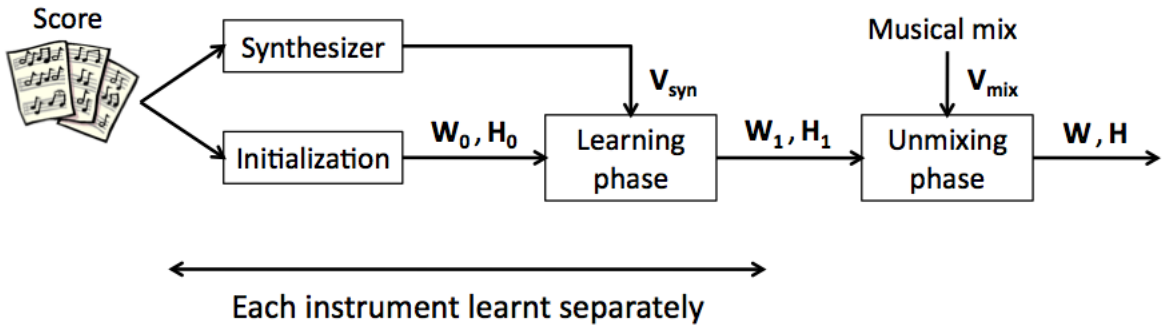


Figure 4: General framework of the improved method

We provide an example of the evolution of the basis function and the amplitude of a single component representing a clarinet note in Figure 5. We plot the harmonic comb and the binary function used for the initialization of the learning phase (top), and their evolution after the so-called learning phase (centre) and the unmixing phase (bottom).

As we can see, the amplitude learnt from the synthetic signal is not very relevant compared to the one obtained from the actual signal after the unmixing phase. The spectral envelope learnt, on the other hand, is much more relevant as it represents a clarinet spectrum similar to the one extracted after the unmixing phase, with a predominance of the harmonics 1 and 3.

This example highlights the asset of the score synthesis method, which provides data close to the physical reality of the instruments and helps to supervise the decomposition of the musical mixture, especially for the basis functions of the dictionary $\mathbf{W}$.

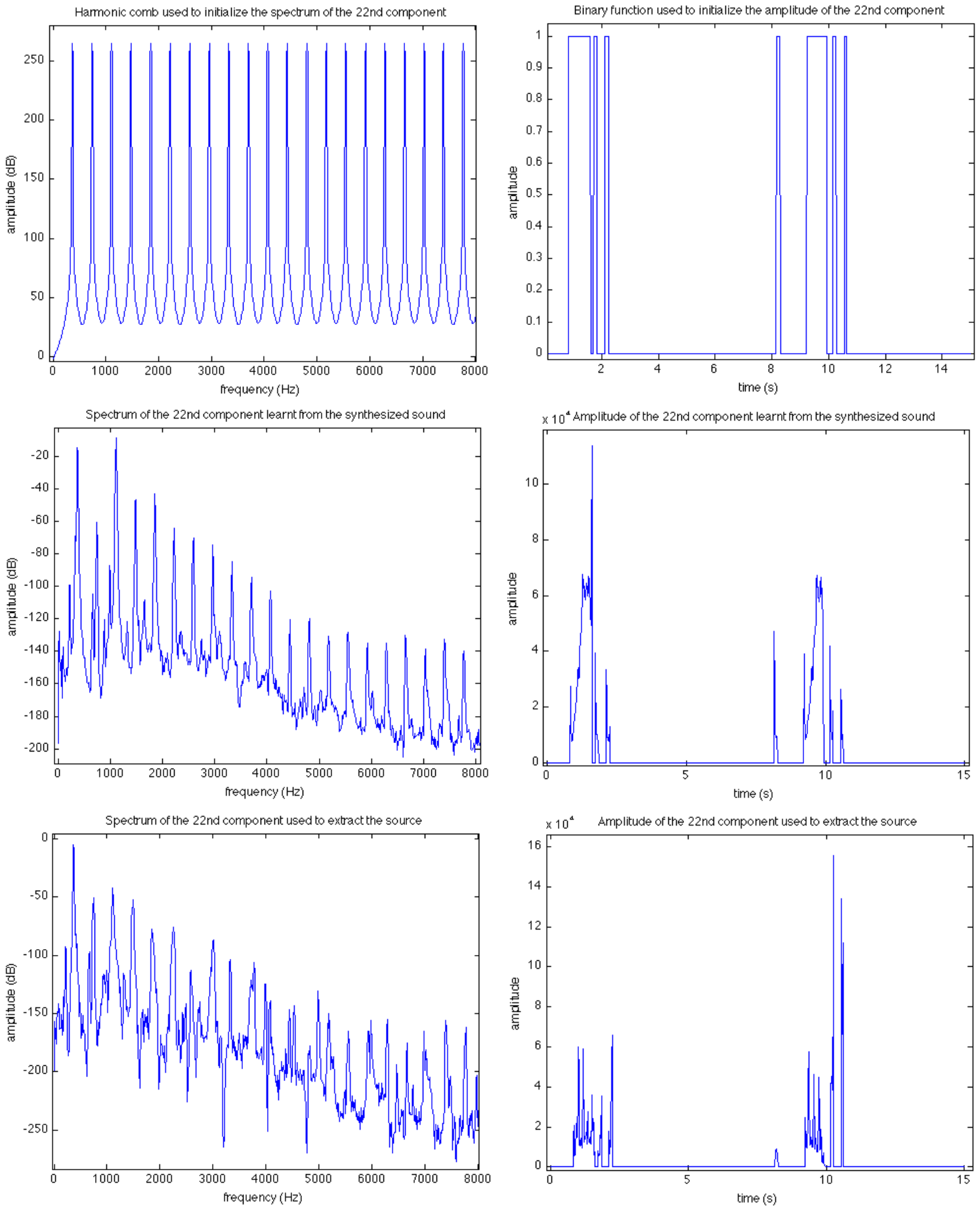


Figure 5: Evolution of the basis function (left) and the amplitude (right) of a component representing a clarinet note during the overall decomposition process

3 Separation results with the proposed method

3.1 The evaluation metrics

The natural way to assess the quality of a source separation algorithm or technique is of course to listen to the extracted sounds and to evaluate them with our own subjective criteria. But in order to obtain exportable results and to allow a fair comparison between different methods, we need to compute some objective evaluation metrics.

Different approaches have been proposed to calculate these evaluation metrics, all based on the comparison of the extracted sources with the original ones. The advantage of these evaluation techniques is to be completely impartial and to provide interpretable metrics, but their inconvenient side is to require the original separated sources of the signals we intend to decompose, hence reducing dramatically the data usable for experiments.

We will now present the two widespread evaluation toolboxes used to assess our proposed method, and explain briefly the calculation of their different metrics.

The BSS_EVAL toolbox

The BSS_EVAL (standing for Blind Source Separation Evaluation) toolbox allows to compute the performance measures elaborated in [18]. In their approach, the authors compare each estimated sources $\widehat{s_j}$ to the given true sources s_j, by decomposing the estimated sources as following

$$\widehat{s_j} = s_{\text{target}} + e_{\text{interf}} + e_{\text{noise}} + e_{\text{artif}}. \tag{27}$$

In this decomposition, $s_{\text{target}} = f(s_j)$ is a version of s_j modified by an allowed distortion f, representing the part of $\widehat{s_j}$ perceived as coming from the wanted source s_j. The terms e_{interf}, e_{noise} and e_{artif} are on their side the interferences, noise and artifacts error terms, and represent the parts of $\widehat{s_j}$ coming from the unwanted sources, from sensor noises and from other artifacts, respectively. The decomposition of the estimated sources into these four measures is achieved through orthogonal projections, which calculations are detailed in [18].

The performance measures are then defined as energy ratios with relevant interpretability, expressed in (dB). The authors define the Source to Distortion Ratio

$$\text{SDR} := 10\log_{10}\frac{||s_{\text{target}}||^2}{||\widehat{s_j} - s_{\text{target}}||^2} \tag{28}$$

the Source to Interference Ratio

$$\text{SIR} := 10\log_{10}\frac{||s_{\text{target}}||^2}{||e_{\text{interf}}||^2} \tag{29}$$

and the Source to Artifacts Ratio

$$\text{SAR} := 10\log_{10}\frac{||\widehat{s_j} - e_{\text{artif}}||^2}{||e_{\text{artif}}||^2}. \tag{30}$$

These three performance measures are inspired from the usual definition of the Signal to Noise Ration (SNR), with few modifications. Their interpretability is quite intuitive, and more information can be found about them in [18].

The PEASS toolbox

The PEASS (standing for Perceptual Evaluation methods for Audio Source Separation) toolbox provides a set of performance measures similar to those from the BSS_EVAL toolbox, with also additional metrics in the form of perceptually-motivated scores rather than energy ratios [19]. In this new approach, the estimated sources $\widehat{s}_j$ are decomposed in a similar manner to the decomposition (27)

$$\widehat{s}_j - s_j = e_{\text{target}} + e_{\text{interf}} + e_{\text{artif}} \tag{31}$$

where the terms e_{target}, e_{interf} and e_{artif} denote the target distortion component, the interference component and the artifacts component, respectively. These three components are calculated through a complex algorithm, approximating inter alia the auditory time-frequency resolution [19].

As mentioned above, the specificity of the PEASS toolbox is then to provide perceptually-motivated scores from these components, in addition to the classic energy ratios. These new performance measures are:

- the Overall Perceptual Score (OPS),
- the Target-related Perceptual Score (TPS),
- the Interference-related Perceptual Score (APS),
- the Artifacts-related Perceptual Score (APS).

These scores are obtained by assessing the salience of each distortion component separately, using the perceptual similarity measure (PSM) provided by the PEMO-Q auditory model [20]. For this, the estimated sources are compared with themselves minus the considered distortions, leading to the following salience features

$$q_j^{\text{overall}} = \text{PSM}(\widehat{s}_j, s_j) \tag{32}$$
$$q_j^{\text{target}} = \text{PSM}(\widehat{s}_j, \widehat{s}_j - e_{\text{target}}) \tag{33}$$
$$q_j^{\text{interf}} = \text{PSM}(\widehat{s}_j, \widehat{s}_j - e_{\text{interf}}) \tag{34}$$
$$q_j^{\text{interf}} = \text{PSM}(\widehat{s}_j, \widehat{s}_j - e_{\text{artif}}). \tag{35}$$

At last, these salience features are combined by a nonlinear mapping, itself adapted to match the subjective grading scale estimated from preliminary perceptual experiments [19]. This method allows to provide the perceptual scores presented above (in %), complementary to the classic energy ratios (in dB).

3.2 Experiments and evaluation

3.2.1 Description of the dataset

We assess our proposed method on the MIREX multi-F0 development set, used in [21] and available through the C4DM Research Data Repository[1]. This dataset consists of a multi-track recording of an extract from a string quartet by Ludwig van Beethoven (op.18 n.5, III. *Andante Cantabile*, var.V.), arranged for a woodwind quintet (flute, oboe, clarinet, French horn and bassoon).

[1] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/12`

This recording originates from the MIREX 2007 F0-tracking competition, and MIDI annotations of each separated tracks have been created and aligned afterwards with the software Sonic Visualizer [22]. The resulting dataset thus contains the `.wav` file (16 bits, 44.1kHz) and the `.mid` file of each instrument, and for the needs of our proposed method we add the `.wav` files of the synthesized signals obtained from the EIC2 sample-based synthesizer integrated in Ableton Live.

3.2.2 Experimental setup and results

We apply our improved method presented in Section 2.3 to the first 15 seconds of the recording, and we compare the separation results with those obtained from our original method presented in Section 2.2. In order to underscore the contribution of the harmonic constraint in the original method, we also use a limited version of this latter, with solely the temporal constraint presented in Section 2.2.1. These three methods are henceforth referred to as "improved", "original" and "t.c. only" (for "temporal constraint only").

For the experiment we use the Itakura-Saito NMF on the power spectrogram, with the Maximization-Minimization (MM) algorithm presented in [6] (which only differs from the algorithm presented in Section 1.1.3 by the use of an exponent $\gamma(\beta)$ in the update equations). The spectrogram is calculated with a 4096-point (93 ms) Hanning window and with 87.5% overlap. In the pianoroll representation of the score we add 100 ms before and 200 ms after each notes, for the reasons mentioned in Section 2.2.1. The number of harmonics in the harmonic model is for its part set to 50 for each instrument.

For each methods we add 30 extra-components with random initializations to collect the residual sounds, for every NMF routines. Finally, the "t.c. only" and "original" methods are run with 30 iterations for the decomposition process, and the "improved" method is run with 15 iterations for the learning phase and 10 iterations for the unmixing phase, as this gives better results in a significant way[1].

The performance measures obtained with the BSS_EVAL and the PEASS toolboxes are presented in Table 1, and the corresponding extracted sounds are available on the C4DM Research Data Repository[2].

From these experimental results, we notice a substantial enhancement of the quality of separation between the "t.c. only" and the "original" methods, with for instance an increase of 10.03 dB in average for the SDR value. The OPS is likewise better for each instrument, except for the flute (which is quite surprising in regards to the extracted sounds).

In the same way, we notice a general improvement between the "original" and the "improved" methods, especially for the clarinet and the oboe. In the selected musical extract, these two instruments are playing the same melody in an octave interval. This specific result, in parallel with the extracted sounds (where the two instruments are mingled with the "original" method and well segregated with the "improved" method) demonstrates the benefit of using a score synthesis method.

[1] This statement will be proved in Section 3.2.3.

[2] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/26`

		BSS_EVAL 3.0			PEASS 2.0			
	method	SDR (dB)	SIR (dB)	SAR (dB)	OPS (%)	TPS (%)	IPS (%)	APS (%)
bassoon	t.c. only	0.83	4.66	4.43	21.79	36.40	14.14	**55.16**
	original	10.35	18.30	11.17	26.69	**45.67**	37.71	46.98
	improved	**11.96**	**20.05**	**12.74**	**27.98**	33.52	**63.84**	28.23
clarinet	t.c. only	1.09	4.81	4.74	16.39	11.24	7.57	31.42
	original	8.74	11.21	12.69	17.57	24.57	10.97	**42.44**
	improved	**14.22**	**22.99**	**14.86**	**26.31**	**44.50**	**34.22**	30.01
flute	t.c. only	3.97	11.41	5.13	24.34	29.32	43.89	**37.80**
	original	14.03	**22.08**	14.80	8.54	**94.54**	20.08	7.89
	improved	**16.51**	21.93	**18.01**	**36.68**	32.81	**61.12**	29.69
horn	t.c. only	−1.23	1.63	4.21	18.16	12.78	18.16	36.05
	original	10.29	18.30	11.10	27.62	41.83	32.48	**48.51**
	improved	**11.17**	**20.64**	**11.73**	**37.72**	**48.64**	**47.14**	48.41
oboe	t.c. only	−9.72	−6.74	0.90	14.62	9.12	8.58	**30.35**
	original	1.66	11.53	2.43	18.60	5.85	**51.91**	9.86
	improved	**7.78**	**16.78**	**8.45**	**25.45**	**40.34**	32.34	17.47

Table 1: Separation results of the three different methods applied to the same extract from the woodwind quintet recording. Best results are shown boldfaced.

Finally, we remark that the perceptual scores provided from the PEASS toolbox are not always correlated to their corresponding energy ratios, with for example the SAR and the APS of the bassoon which seems to evolve in the opposite direction depending on the method used.

3.2.3 Influence of the various parameters

An observation brought out by the many experiments run with the "improved" method is that the separation results highly depend on the various parameters of the general decomposition process, such as the parameters of the NMF algorithm, of the spectrogram or of the temporal and harmonic constraints.

As mentioned above, the number of iterations of the unmixing phase appeared to be one of most influential parameters. We could have thought that the separation results would increase with the number of iterations, as for many other source separation algorithms, but in our case the reality seemed to be different.

In order to investigate the influence of this specific parameter, we set a new experiment comparing the separation results with a number of iterations for the unmixing phase going from 0 to 100. In this experiment, we also vary the exponent of the spectrogram (magnitude or power) and the β-divergence used (Itakura-Saito, Kullback-Leibler or Euclidean distance) to study their own weight on the separation results. All the other parameters are maintained constant, with the same values as in the first experiment.

As it would be difficult to exhibit here the seven different performance measures resulting from this experiment, only the evolution of the global SDR (the average of the different SDR values) is presented in Figure 6.

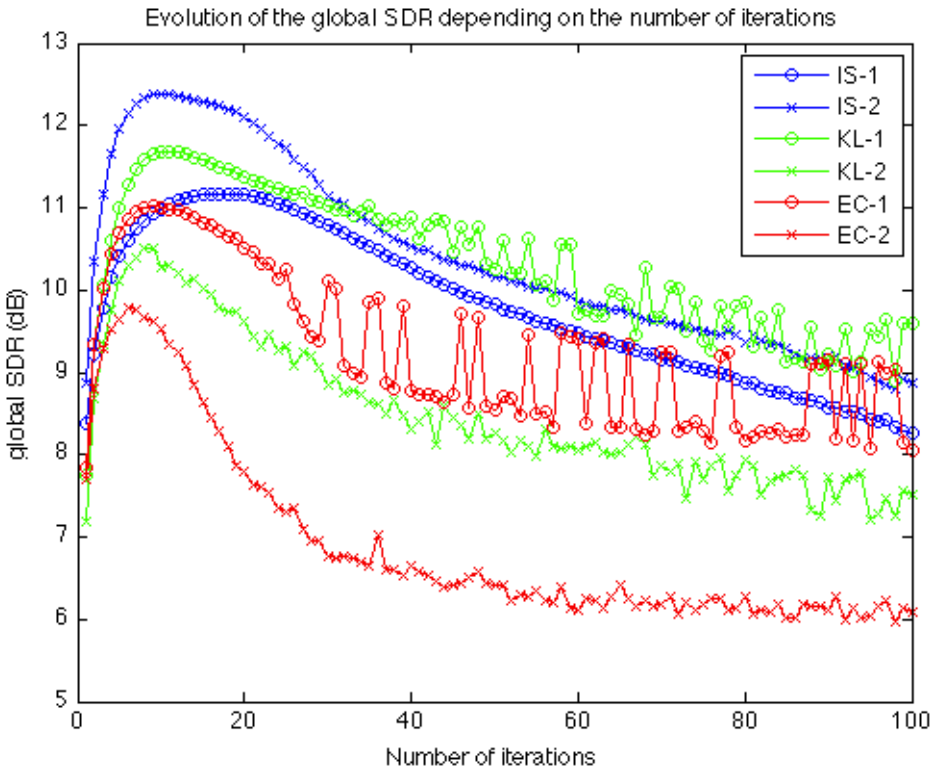


Figure 6: Evolution of the global SDR depending on the number of iterations during the unmixing phase, the exponent of the spectrogram (magnitude or power) and the divergence used in the β-NMF algorithm

As we can see, the Itakura-Saito NMF on the power spectrogram and the Kullback-Leibler NMF on the magnitude spectrogram seem to give the best separation results for this example, with an optimal number of iterations for the unmixing phase between 10 and 15 approximately. The interesting fact is that all the curves have a similar behavior, with a fast increase during the first few iterations and a slow lessening after about 15 iterations.

A sensible explanation for this phenomena is that a good compromise to represent the data from the mixture is found between the synthetic model (0 iteration) and what could be called an "overlearnt" version of the decomposition (20 iterations and more). In that later case, the distance between the spectrogram $\mathbf{V_{mix}}$ and its approximation $\mathbf{WH}$ keeps decreasing, but the quality of separation gets worse as the harmonic constraint becomes weaker and the sources are more likely to "leak" on one another.

This experiment allows to validate the importance given to choice of the various parameters during the decomposition process, as it demonstrates their great influence on the separation results. The best combination seems to be the one used in the first experiment, *i.e.* the Itakura-Saito NMF on the power spectrogram, with 10 iterations for the unmixing phase. It is difficult though to discern the parts of these observations coming from the proposed method and those coming from the musical example used. For this, we would need to apply our method to other recordings in order to determine the general characteristics of the method and the features specific to each examples, as we will see in Section 3.3.1.

3.2.4 Comparison with another method

We compared our proposed method with another score-informed source separation method from the literature, namely an adapted version of [15]. This resulted in a paper entitled "A Comparison of Two Different Methods for Score-Informed Source Separation", accepted to the 5th International Workshop on Machine Learning and Music (MML12) held in conjunction with the International Conference on Machine Learning (ICML 2012) in Edinburgh at the end of June. A copy of this paper can be found in the Appendix A of this thesis, and the data attached to it (including the code used for comparison and the extracted sounds) is available through the C4DM Research Data Repository[1].

The experimental setup used to compare the two methods was identical to the one presented in Section 3.2.2, and the musical extract employed was evenly the 15 first seconds of the woodwind quintet from the MIREX dataset. In our case we used the Itakura-Saito divergence on the power spectrogram.

The PLCA-based method of [15] was only adapted to have the same extraction method as the one presented in Section 1.2.2, based on Wiener filtering. Both methods were run with 30 iterations for the learning phase, and our proposed method (referred to as "Method A" in the paper) used 10 iterations for the unmixing phase while the method from [15] (referred to as "Method B" in the paper) used 20 of them.

The performance measures from this comparison are presented in Table 2, with the mean metrics from the BSS_EVAL and PEASS toolboxes calculated over 100 runs. The results obtained with our proposed method (A-10 here) are slightly better than those obtained previously ("improved" rows in Table 1), due to the use of a corrected function to create the harmonic combs and to the greater number of iterations in the learning phase.

From these measures, we observe that our method gives overall better separation results for this specific example, especially for the French horn and the oboe. This is very likely to be due to the temporal and harmonic constraints than our method incorporates in the decomposition process.

We also notice from the standard deviation (see Table 2) that our method has a more stable behavior, with a standard deviation inferior to $\pm 0.03\%$ in the BSS_EVAL metrics and inferior to $\pm 0.73\%$ in the PEASS metrics. This can be explained by the lower influence of the random initializations in our method, where only the extra-components used to collect the residual sounds are initialized in such a way, unlike the other method where it is the case for all the components.

3.3 Creation of a new dataset

3.3.1 Motivation and presentation

As mentioned above, it is not relevant to assess the quality of our source separation method on a single musical example. But the problem is that the available datasets with score-aligned multitrack recordings are very few, if not non-existent.

[1] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/20`

		BSS_EVAL 3.0			PEASS 2.0			
	method	SDR (dB)	SIR (dB)	SAR (dB)	OPS (%)	TPS (%)	IPS (%)	APS (%)
bassoon	A-10	$\mathbf{11.97_{\pm 0.01}}$	$\mathbf{20.43_{\pm 0.01}}$	$\mathbf{12.67_{\pm 0.01}}$	$27.39_{\pm 0.09}$	$32.27_{\pm 0.58}$	$\mathbf{62.16_{\pm 0.37}}$	$27.42_{\pm 0.31}$
	B-20	$10.68_{\pm 0.32}$	$18.82_{\pm 0.75}$	$11.47_{\pm 0.30}$	$\mathbf{33.08_{\pm 0.95}}$	$\mathbf{35.06_{\pm 2.10}}$	$57.98_{\pm 1.75}$	$\mathbf{37.76_{\pm 1.47}}$
clarinet	A-10	$\mathbf{14.45_{\pm 0.01}}$	$\mathbf{23.46_{\pm 0.02}}$	$\mathbf{15.06_{\pm 0.01}}$	$\mathbf{26.33_{\pm 0.49}}$	$\mathbf{41.61_{\pm 0.35}}$	$\mathbf{33.04_{\pm 0.59}}$	$\mathbf{30.58_{\pm 0.42}}$
	B-20	$11.92_{\pm 0.48}$	$17.42_{\pm 0.74}$	$13.45_{\pm 0.48}$	$14.81_{\pm 1.85}$	$25.74_{\pm 2.95}$	$14.93_{\pm 2.29}$	$25.67_{\pm 2.89}$
flute	A-10	$\mathbf{16.51_{\pm 0.00}}$	$\mathbf{22.11_{\pm 0.01}}$	$\mathbf{17.94_{\pm 0.01}}$	$\mathbf{37.41_{\pm 0.22}}$	$\mathbf{35.68_{\pm 0.71}}$	$\mathbf{60.41_{\pm 0.72}}$	$30.77_{\pm 0.66}$
	B-20	$12.49_{\pm 0.57}$	$21.86_{\pm 0.55}$	$13.05_{\pm 0.59}$	$32.18_{\pm 1.11}$	$31.56_{\pm 2.25}$	$51.91_{\pm 2.92}$	$\mathbf{33.94_{\pm 2.19}}$
horn	A-10	$\mathbf{11.10_{\pm 0.01}}$	$\mathbf{20.96_{\pm 0.02}}$	$\mathbf{11.61_{\pm 0.02}}$	$\mathbf{37.76_{\pm 0.29}}$	$49.30_{\pm 0.33}$	$\mathbf{47.84_{\pm 0.52}}$	$49.10_{\pm 0.24}$
	B-20	$5.03_{\pm 0.45}$	$8.27_{\pm 0.65}$	$8.44_{\pm 0.23}$	$11.41_{\pm 0.79}$	$\mathbf{62.71_{\pm 4.90}}$	$2.87_{\pm 0.81}$	$\mathbf{66.85_{\pm 4.10}}$
oboe	A-10	$\mathbf{7.93_{\pm 0.01}}$	$\mathbf{17.60_{\pm 0.02}}$	$\mathbf{8.50_{\pm 0.01}}$	$\mathbf{26.58_{\pm 0.30}}$	$40.50_{\pm 0.27}$	$\mathbf{33.95_{\pm 0.41}}$	$17.83_{\pm 0.31}$
	B-20	$-0.52_{\pm 1.07}$	$1.92_{\pm 1.54}$	$5.46_{\pm 0.61}$	$25.03_{\pm 1.18}$	$\mathbf{42.01_{\pm 2.63}}$	$16.37_{\pm 1.71}$	$\mathbf{57.50_{\pm 1.01}}$

Table 2: Separation results obtained from the comparison of our proposed method (A-10) with an adapted version of [15] (B-20). Mean metrics are calculated over 100 runs, with standard deviation in subscript. Best results are shown boldfaced.

In order to study the influence of the various parameters of our method on other musical examples, we thus decided to create our own dataset of annotated multi-track recordings.

This dataset is compound of five short extracts from chamber music trio pieces, and has therefore been called "TRIOS". Each separated instrumental track is provided with a manually-aligned version of the corresponding score in a MIDI format. The five pieces of music in question are:

- a trio for clarinet, viola and piano by Wolfgang A. Mozart (K.498)
- a trio for violin, cello and piano by Franz Schubert (D.929, op.100)
- a trio for violin, French horn and piano by Johannes Brahms (op.40)
- a trio for trumpet, bassoon and piano by Mathieu Lussier (op.8)
- a trio version of "Take Five" by Paul Desmond, for alto sax, piano and drums

The complete "information sheet" of the TRIOS dataset can be found in the Appendix B of this thesis, and the actual dataset is available through the C4DM Research Data Repository[1].

3.3.2 Data generation

The separated tracks and the aligned MIDI scores from the TRIOS dataset are created and edited as following. First, the original MIDI scores are downloaded from the Kunst der Fuge database[2] or generated from the music edition software Sibelius, and then imported in the sequencer Ableton Live.

The different tracks are then recorded separately, while the musicians listen to the other synthesized parts synchronized with a metronome through headphones. The recordings are afterwards edited and mixed in Digital Performer, and the MIDI scores are eventually manually aligned one by one with Sonic Visualizer.

[1] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/27`

[2] `http://www.kunstderfuge.com/`

3.4 Last improvements and results

Before applying our proposed method to our new dataset, we attempt to make some improvements to it. First of all, the harmonic constraint incorporated in the method is not suited for non-harmonic instruments, such as drums or percussions. We fix this issue by adding a "harmonic/percussive" parameter in the function generating the spectral model of each instruments during the initialization of the learning phase. In the "percussive" case, the components of the instrument are initialized with uniform distributions, similar to those used for the extra-components collecting the residual sounds.

We try next to separate the residual sounds contained in the additional "extra-source", in order to associate the non-harmonic contributions of each sources to their respective harmonic parts and to obtain therefore a complete separation of the mixture. For this we run another NMF routine on the extracted spectrogram corresponding to the "extra-source", with random initializations for the basis functions and with the pianoroll of the score for the initialization of the activation coefficients. This method does not provide satisfying results though and is thus not retained.

Finally, we incorporate a smoothness criteria on the activation coefficients of the sustained notes, in order to obtain better perceptual results.

3.4.1 Incorporation of a smoothness criteria

We use the smoothness criteria on the activation coefficients of $\mathbf{H}$ presented in [23] and integrated by the author in the β-NMF algorithm of Section 1.1.3 with the Itakura-Saito divergence.

This smoothness criteria consists of incorporating the following penalty term

$$P(\mathbf{H}) = \sum_{k=1}^{K} \sum_{n=2}^{N} d_{IS}(\mathbf{H}_{k,(n-1)} | \mathbf{H}_{k,n}) \tag{36}$$

in the cost function (9), such that this latter becomes

$$C_{IS} = D_{IS}(\mathbf{V}|\mathbf{WH}) + \lambda P(\mathbf{H}). \tag{37}$$

This additional term $P(\mathbf{H})$ penalizes large deviations between two consecutive activation coefficients $\mathbf{H}_{k,n}$ and $\mathbf{H}_{k,(n-1)}$, as measured by the IS divergence. It enforces thus the amplitude of each component to observe more or less smooth variations, depending on the positive scalar λ representing the penalty weight.

The gradient descent algorithm presented in Section 1.1.3 applied to the cost function (37) leads to the following multiplicative rule for a single coefficient $\mathbf{H}_{\tilde{k},\tilde{n}}$

$$\mathbf{H}_{\tilde{k},\tilde{n}} \leftarrow \tilde{\mathbf{H}}_{\tilde{k},\tilde{n}} \cdot \frac{\sum_{f=1}^{F} \mathbf{V}_{f,\tilde{n}} \hat{\mathbf{V}}_{f,\tilde{n}}^{-2} \mathbf{W}_{f,\tilde{k}} + \lambda \mathbf{H}_{\tilde{k}(\tilde{n}-1)}}{\sum_{f=1}^{F} \hat{\mathbf{V}}_{f,\tilde{n}}^{-1} \mathbf{W}_{f,\tilde{k}} + \lambda \mathbf{H}_{\tilde{k}(\tilde{n}+1)}^{-1}} \tag{38}$$

which cannot be expressed in a convenient matrix form similar to (22), but can still be vectorized in order to provide the simple implementation given in [23].

This smoothness criteria can enhance the perceptual quality of separation for sustained notes, by preventing their amplitudes to flicker (as it can be the case in the unpenalized case).

In order to penalize these sustained notes only, we just have to replace the penalty weight λ by λ_{kn} in the multiplicative update rule (38). This requests however to create a matrix $\boldsymbol{\lambda}$ of the same dimensions as $\mathbf{H}$, whose entries are equal to 0 except for the concerned components and time frames. This matrix can be generated automatically by retrieving the sustained notes in the pianoroll representation of the score.

In our case, we simply specify the notes that we intend to penalize by replacing the parameter λ by λ_k in the multiplicative update rule (38), as we have a decomposition form of one component per note.

3.4.2 Evaluation with the new dataset

We can finally apply our proposed method to the new dataset, in order to assess its quality of separation on the five musical examples. At first, we run the experiment presented in Section 3.2.3. on the the five recordings, with the same experimental setup. The only difference here is that we keep the entire duration of the signals, going from 18 to 53 seconds depending on the extracts.

The results of this new experiment are presented in Figure 7 for each example[1]. We notice that for all these musical examples, the "over-learning" phenomena observed with the woodwind quintet is evenly present. The global SDR values are on the whole better with a number of iterations for the unmixing phase between 10 and 15, and they all decrease progressively passed this stage (except for the IS divergence on the power spectrogram case in the Take Five example, where the SDR value remains constant).

By cons, the configuration that seems to give the best results on average is not the IS divergence on the power spectrogram, as it was suggested by the woodwind quintet example, but the KL divergence on the magnitude spectrogram. The confrontation between these two configurations is widely discussed in the literature, as in [6] for example, and would need a further investigation in the present case.

Given these observations, we decide to provide the performance measures obtained from the BSS_EVAL toolbox and calculated on the five musical examples, with the KL divergence on the magnitude spectrogram and with 10 iterations for the unmixing phase. These results are presented in Table 3, and can be used for further comparisons as the performance level of our proposed method. The resulting extracted sounds are also available on the C4DM Research Data Repository[2].

Finally, we attempt to highlight the gain obtained with the smoothness criteria presented above with one last experiment. We compare the measures obtained from the BSS_EVAL and the PEASS toolboxes when a penalty weight of $\lambda = 1, 10$ and 100 respectively is used for the French horn in the woodwind quintet example.

[1] The measures with the Euclidean distance for the Take Five example have not been calculated by the evaluation toolbox, for unknown reasons.

[2] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/26`

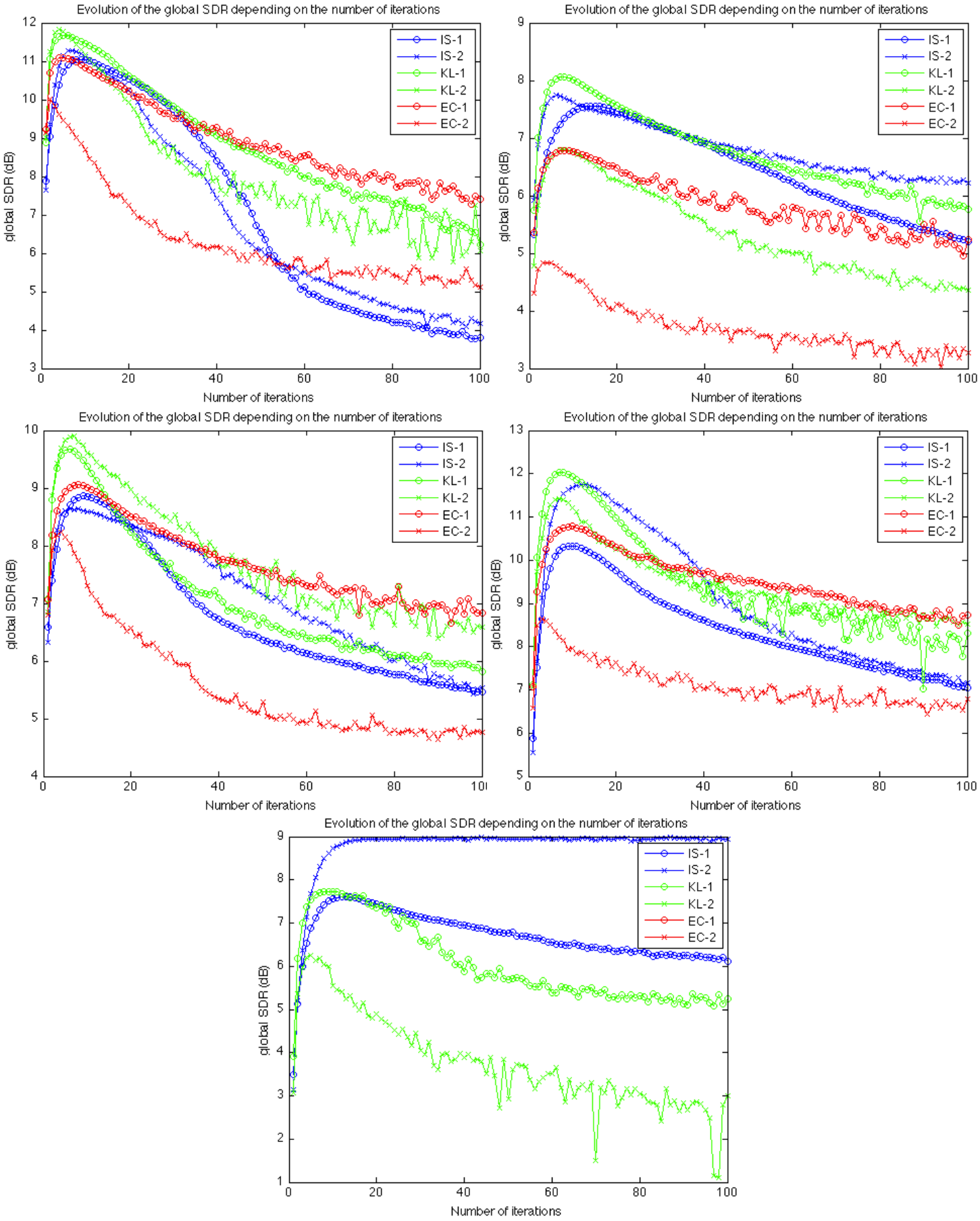


Figure 7: Evolution of the global SDR for the Mozart (top left), Schubert (top right), Brahms (middle left), Lussier (middle right) and Take 5 (bottom) examples

recording	instrument	BSS_EVAL 3.0 SDR (dB)	SIR (dB)	SAR (dB)
Mozart	clarinet	14.67	22.46	15.49
	viola	4.54	11.33	5.86
	piano	8.91	13.09	11.20
Schubert	violin	11.67	21.39	12.19
	cello	10.92	19.50	11.62
	piano	13.20	18.29	14.88
Brahms	violin	11.91	21.22	12.49
	horn	12.30	21.13	12.94
	piano	10.17	14.61	12.25
Lussier	trumpet	9.72	18.84	10.52
	bassoon	8.09	13.41	9.80
	piano	6.22	9.88	9.09
Take Five	alto sax	13.94	22.87	14.56
	piano	8.69	12.33	11.39
	drums (av.)	5.33	15.99	6.70
	kick	12.03	29.15	12.12
	ride	0.64	4.64	4.13
	snare	3.31	14.17	3.84

Table 3: Separation results given by our proposed method applied to the five recordings from the TRIOS dataset, with the Kullback-Leibler divergence on the magnitude spectrogram and with 10 iterations for the unmixing phase.

This instrument is indeed playing two very long notes in the considered extract and their amplitude is flickering in the unpenalized case, which can be unpleasant whilst listening to the corresponding extracted sound. In order to penalize these few notes only during the decomposition process, we apply a penalty weight λ_k on the corresponding components, as explained in Section 3.4.1.

The results from this experiment are presented in Table 4, and the corresponding extracted sounds are once again available on the C4DM Research Data Repository[1]. From these results, we observe that the SDR, the OPS and the IPS values increase with the use of a higher penalty weight, but it is not the case for the other metrics. The difference is however difficult to perceive whilst listening to the extracted sounds, and again this experiment would need further investigation.

	method	BSS_EVAL 3.0 SDR (dB)	SIR (dB)	SAR (dB)	PEASS 2.0 OPS (%)	TPS (%)	IPS (%)	APS (%)
horn	unpenalized	11.10	20.30	11.69	36.93	**49.48**	45.59	**49.95**
	$\lambda = 1$	11.15	20.32	11.75	37.23	49.45	46.09	49.74
	$\lambda = 10$	11.34	**20.35**	**11.97**	36.89	49.37	45.80	49.41
	$\lambda = 100$	**11.83**	20.22	11.69	**38.27**	47.33	**46.86**	47.77

Table 4: Separation results obtained by applying a smoothness criteria on the French horn in the woodwind quintet example. Best results are shown boldfaced.

[1] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/26`

Conclusion

In this thesis, we have presented an efficient method for score-informed source separation. Its specificity is to use a decomposition form of one component per note per instrument in a NMF framework, and to inform the features of these components with a constrained learning phase on signals previously synthesized from the score.

This method produces very acceptable audio results, and appeared to give better performance measures than another method from the literature. Its strength is to supervise the evolution of the activation coefficients and the basis functions of each instruments during the overall decomposition process, thanks to the temporal and harmonic constraints used. Its weak point however is to collect all the residual sounds in an single extra-source, due to the use of the so-called harmonic constraint.

This feature is more or less inconvenient, depending on the application of the source separation method. For "desoloing" applications for example, it is not acceptable to withdraw only the harmonic part of an instrument and to leave its non-harmonic part in the mixture. But for remixing application, it is not a big issue to amplify or to lessen only the harmonic contribution of one or more instruments.

A concrete application of our method could therefore be to create an "instrument equalizer", where the volume of each instrument would be slightly adjustable in the manner of the frequency bands in a classic equalizer. But the aligned score required by the method is usually long and complicated to obtain, and so this "instrument equalizer" would rather be applicable on valuable recordings only, as for the restoration of historical records for instance.

References

[1] C. Févotte. Itakura-Saito nonnegative factorizations of the power spectrogram for music signal decomposition. In Wenwu Wang, editor, *Machine Audition: Principles, Algorithms and Systems*, chapter 11. IGI Global Press, 2010.

[2] P. Paatero. Least squares formulation of robust non-negative factor analysis. *Chemometrics and Intelligent Laboratory Systems*, 37(1):23–25, 1997.

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

[4] A. Basu, I. R. Harris, N. L. Hjort, and M. C. Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

[5] S. Eguchi and Y. Kano. Robustifying maximum likelihood estimation. Technical report, Institute of Statistical Mathematics, 2001.

[6] C. Févotte and J. Idier. Algorithms for nonnegative matrix factorization with the beta-divergence. *Neural Computation*, 23(9):2421–2456, 2011.

[7] H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for multiple-template shift-variant non-negative matrix deconvolution based on β-divergence. Technical Report C4DM-TR-06-12, Queen Mary University of London, 2012. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-06-12`.

[8] H. Kirchhoff, S. Dixon, and A. Klapuri. Multi-template shift-variant non-negative matrix deconvolution for semi-automatic music transcription. In *13th International Conference on Music Information Retrieval*, Porto, 2012.

[9] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. *Signal Processing*, 57(3):177–180, 2003.

[10] B. Wang and M. D. Plumbley. Musical audio stream separation by non-negative matrix factorization. *Electronic Engineering*, 38(3):206–210, 2005.

[11] R. Hennequin. *Décomposition de spectrogrammes musicaux informée par des modèles de synthèse spectrale. Modélisation des variations temporelles dans les éléments sonores.* PhD thesis, Télécom Paristech, 2011.

[12] J. Woodruff, B. Pardo, and R. Dannenberg. Remixing stereo music with score-informed source separation. In Kjell Lemstrom, Adam Tindale, and RogerEditors Dannenberg, editors, *Electrical Engineering*, pages 314–319. University of Victoria, 2006.

[13] C. Raphael. A classifier-based approach to score-guided source separation of musical audio. *Computer Music Journal*, 32(1):51–59, 2008.

[14] P. Smaragdis and G. Mysore. Separation by "humming": User-guided sound extraction from monophonic mixtures. *WASPAA 09*, pages 69–72, 2009.

[15] J. Ganseman, G. Mysore, P. Scheunders, and J. Abel. Source separation by score synthesis. In *Proc of the International Computer Music Conference*, pages 462–465, New York, NY, USA, 2010.

[16] R. Hennequin, B. David, and R. Badeau. Score informed audio source separation using a parametric model of non-negative spectrogram. In *Proc of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 45–48, Prague, Czech Republic, 2011.

[17] Z. Duan and B. Pardo. Soundprism: An online system for score-informed source separation of music audio. *J. Sel. Topics Signal Processing*, 5(6):1205–1215, 2011.

[18] E. Vincent, R. Gribonval, and C. Févotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech and Language Processing*, 14(4):1462–1469, 2006.

[19] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann. Subjective and objective quality assessment of audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2046–2057, 2011.

[20] R. Huber and B. Kollmeier. PEMO-Q - A new method for objective audio quality assessment using a model of auditory perception. *IEEE Transactions on Audio, Speech and Language Processing*, 14(6):1902–1911, 2006.

[21] E. Benetos and S. Dixon. Joint multi-pitch detection using harmonic envelope estimation for polyphonic music transcription. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1111–1123, 2011.

[22] C. Cannam, C. Landone, and M. Sandler. Sonic visualiser: an open source application for viewing, analysing, and annotating music audio files. In *Proceedings of the international conference on Multimedia*, pages 1467–1468, Firenze, Italy, 2010.

[23] C. Févotte. Majorization-minimization algorithm for smooth Itakura-Saito nonnegative matrix factorization. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, 2011.

A Paper accepted to MML12

A Comparison of Two Different Methods for Score-Informed Source Separation

Joachim Fritsch JOACHIM.FRITSCH@ATIAM.FR
Master ATIAM, University Pierre and Marie Curie, 4 place Jussieu, 75005 Paris, France

Joachim Ganseman JOACHIM.GANSEMAN@UA.AC.BE
IBBT-Visionlab, University of Antwerp, Universiteitsplein 1, 2610 Antwerp, Belgium

Mark D. Plumbley MARK.PLUMBLEY@EECS.QMUL.AC.UK
Centre for Digital Music, Queen Mary University of London, Mile End Road, London E1 4NS, UK

Abstract

We present a new method for score-informed source separation, combining ideas from two previous approaches: one based on parametric modeling of the score which constrains the NMF updating process, the other based on PLCA that uses synthesized scores as prior probability distributions. We experimentally show improved separation results using the BSS_EVAL and PEASS toolkits, and discuss strengths and weaknesses compared with the previous PLCA-based approach.

1. Introduction

Musical audio source separation seeks to isolate the different instruments in a musical mixture. Many approaches have been proposed in order to conduct this separation, of which those using Nonnegative Matrix Factorization (NMF) and Probabilistic Latent Component Analysis (PLCA) have been shown to be effective.

More recently, the use of information from musical scores has been addressed to guide these algorithms and to improve the quality of separation. (Ganseman et al., 2010) aligned the synthesized score to the original audio, providing priors to the PLCA decomposition of the mixture. (Hennequin et al., 2011) used the score to initialize an algorithm based on a parametric decomposition of the spectrogram with NMF.

Work partly supported by an IWT Flanders Specialization Grant, EPSRC Leadership Fellowship EP/G007144/1, and EU FET-Open Project FP7-ICT-225913 "SMALL".
In *5th International Workshop on Machine Learning and Music*, Edinburgh, Scotland, UK, 2012.

In this paper we adapt and combine both methods, by synthesizing the score and learning the components of the different instruments separately, and then using the information learnt to initialize the decomposition process with NMF. Our proposed method is presented as 'Method A' and compared with an updated version of (Ganseman et al., 2010), presented as 'Method B'.

2. Score-Informed Source Separation

A musical score provides a wide range of information, such as the pitch, the onset time and the duration of each note played by each instrument. This information can therefore be used to supply spectral and temporal information to the separation algorithm. In this paper we use a perfectly aligned MIDI file and we do not consider the problem of score-to-audio alignment.

2.1. Description of Method A

As in (Ganseman et al., 2010), we initially learn the dictionaries and the activation coefficients of each instrument separately with the synthesized scores.

We use the Itakura-Saito (IS) NMF of the power spectrogram with multiplicative updates (Févotte, 2010), and we initialize the activation coefficients with a 'pianoroll' representation of the score, with one component per note. We also use a harmonic model to initialize the dictionaries, with a harmonic comb adapted to the fundamental frequency of each note (Hennequin et al., 2011). We add about 10 extra components with random initializations, to collect the residual sounds.

We use the information learnt to initialize a second IS-NMF routine on the actual musical mix, adding again about 30 extra components to collect the residual sounds. Finally, we separate the different instru-

		BSS_EVAL 3.0			PEASS 2.0			
	method	SDR (dB)	SIR (dB)	SAR (dB)	OPS (%)	TPS (%)	IPS (%)	APS (%)
bassoon	A-10	**11.97**$_{\pm 0.01}$	**20.43**$_{\pm 0.01}$	**12.67**$_{\pm 0.01}$	27.39$_{\pm 0.09}$	32.27$_{\pm 0.58}$	**62.16**$_{\pm 0.37}$	27.42$_{\pm 0.31}$
	B-20	10.68$_{\pm 0.32}$	18.82$_{\pm 0.75}$	11.47$_{\pm 0.30}$	**33.08**$_{\pm 0.95}$	**35.06**$_{\pm 2.10}$	57.98$_{\pm 1.75}$	**37.76**$_{\pm 1.47}$
clarinet	A-10	**14.45**$_{\pm 0.01}$	**23.46**$_{\pm 0.02}$	**15.06**$_{\pm 0.01}$	**26.33**$_{\pm 0.49}$	**41.61**$_{\pm 0.35}$	**33.04**$_{\pm 0.59}$	**30.58**$_{\pm 0.42}$
	B-20	11.92$_{\pm 0.48}$	17.42$_{\pm 0.74}$	13.45$_{\pm 0.48}$	14.81$_{\pm 1.85}$	25.74$_{\pm 2.95}$	14.93$_{\pm 2.29}$	25.67$_{\pm 2.89}$
flute	A-10	**16.51**$_{\pm 0.00}$	**22.11**$_{\pm 0.01}$	**17.94**$_{\pm 0.01}$	**37.41**$_{\pm 0.22}$	**35.68**$_{\pm 0.71}$	**60.41**$_{\pm 0.72}$	30.77$_{\pm 0.66}$
	B-20	12.49$_{\pm 0.57}$	21.86$_{\pm 0.55}$	13.05$_{\pm 0.59}$	32.18$_{\pm 1.11}$	31.56$_{\pm 2.25}$	51.91$_{\pm 2.92}$	**33.94**$_{\pm 2.19}$
horn	A-10	**11.10**$_{\pm 0.01}$	**20.96**$_{\pm 0.02}$	**11.61**$_{\pm 0.02}$	**37.76**$_{\pm 0.29}$	49.30$_{\pm 0.33}$	**47.84**$_{\pm 0.52}$	49.10$_{\pm 0.24}$
	B-20	5.03$_{\pm 0.45}$	8.27$_{\pm 0.65}$	8.44$_{\pm 0.23}$	11.41$_{\pm 0.79}$	**62.71**$_{\pm 4.90}$	2.87$_{\pm 0.81}$	**66.85**$_{\pm 4.10}$
oboe	A-10	**7.93**$_{\pm 0.01}$	**17.60**$_{\pm 0.02}$	**8.50**$_{\pm 0.01}$	**26.58**$_{\pm 0.30}$	40.50$_{\pm 0.27}$	**33.95**$_{\pm 0.41}$	17.83$_{\pm 0.31}$
	B-20	−0.52$_{\pm 1.07}$	1.92$_{\pm 1.54}$	5.46$_{\pm 0.61}$	25.03$_{\pm 1.18}$	**42.01**$_{\pm 2.63}$	16.37$_{\pm 1.71}$	**57.50**$_{\pm 1.01}$

Table 1. Quality of source separation results of a woodwind quintet. We display mean BSS_EVAL and PEASS metrics calculated over 100 runs, with standard deviation shown in subscript. Method A was run with 10 iterations and method B with 20 iterations in the mixture factorization phase. Higher is better for all scores, best scores are shown boldfaced.

ments with a Wiener masking method (Févotte, 2010).

2.2. Description of Method B

This method (Ganseman et al., 2010) only uses synthesized score parts to learn the dictionary and activation matrices that serve as prior distributions to PLCA. PLCA has been shown to be numerically equivalent to NMF with a Kullback-Leibler divergence. The method does not rely on any MIDI representation, so in the following experiment we apply it with a fixed number of 20 components per source on the magnitude spectrogram. Not anticipating a 6th source, we also do not provide additional components. To allow a fairer comparison, we altered the reconstruction phase to also use the normalized source estimates as a mask on the mixture spectrogram, i.e. Wiener filtering.

3. Results and Conclusion

We apply both methods to the first 15 seconds of the woodwind quintet recording from the MIREX 2007 F0-tracking competition. A 4096-point STFT with 87.5% overlap was used. Scores were synthesized using the EIC2 synthesizer integrated in Ableton Live, and the matrices for initialization (Method A) or prior distributions (Method B) were learnt from those in 30 iterations. Afterwards Method A was run for 10 iterations and Method B for 20 iterations, as this gave good results for each.

We use the BSS_EVAL (Vincent et al., 2006) and PEASS (Emiya et al., 2011) toolboxes for evaluation. The results of our experiment are summarized in table 1. We find that in this example, Method A gives overall better results, due to the harmonic and temporal constraints that Method A incorporates in the update process. The lack of those is likely the cause of Method B to have worse interference-related metrics (SIR, IPS), having more leakage from other sources into the extracted sounds. From the standard deviation measurement, we also notice that Method A has a more stable behavior than Method B. The dataset used for the experiment, the code and the resulting sound files are available through the C4DM Research Data Repository at `http://c4dm.eecs.qmul.ac.uk/rdr/`.

In the future, Method B could be improved by incorporating harmonic and temporal constraints similar to those from Method A. The parametric model of this latter would also need adjustments in the case of inharmonic or percussive sounds.

References

Emiya, V., Vincent, E., Harlander, N., and Hohmann, V. Subjective and objective quality assessment of audio source separation. *IEEE Trans. Audio Speech Lang. Proc.*, 19(7):2046–2057, 2011.

Févotte, C. Itakura-Saito nonnegative factorizations of the power spectrogram for music signal decomposition. In Wang, Wenwu (ed.), *Machine Audition*, chapter 11. IGI Global Press, 2010.

Ganseman, J., Mysore, G., Scheunders, P., and Abel, J. Source separation by score synthesis. In *Proc. ICMC*, pp. 462–465, New York, USA, 2010.

Hennequin, R., David, B., and Badeau, R. Score informed audio source separation using a parametric model of non-negative spectrogram. In *Proc. ICASSP*, pp. 45–48, Prague, Czech Republic, 2011.

Vincent, E., Gribonval, R., and Févotte, C. Performance measurement in blind audio source separation. *IEEE Trans. Audio Speech Lang. Proc.*, 14(4): 1462–1469, 2006.

B The TRIOS dataset information sheet

The TRIOS dataset

Joachim Fritsch
joachim.fritsch@atiam.fr

Master ATIAM
University Pierre and Marie Curie
4 place Jussieu, 75005 Paris, France

1 Introduction

The TRIOS dataset is a score-aligned multitrack recordings dataset which can be used for various research problems, such as Score-Informed Source Separation, Automatic Music Transcription, etc. This dataset consists of the separated tracks from five recordings of chamber music trio pieces, with their aligned MIDI scores.

2 Download

This dataset can be downloaded through the C4DM Research Data Repository at http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/27.

3 How to Cite

If you use the dataset in a work of your own that you wish to publish, please cite the following thesis:

- Joachim Fritsch. High Quality Musical Audio Source Separation. Master's thesis, UPMC / IRCAM / Telecom Paristech, 2012

4 Content

The five recordings are short extract from the following pieces of music:

- a trio for clarinet, viola and piano by Wolfgang A. Mozart (K.498)
- a trio for violin, cello and piano by Franz Schubert (D.929, op.100)
- a trio for violin, French horn and piano by Johannes Brahms (op.40)
- a trio for trumpet, bassoon and piano by Mathieu Lussier (op.8)
- a trio version of "Take Five" by Paul Desmond, for alto sax, piano and drums

For each musical extract, the `.wav` file and the manually-aligned `.mid` file of each instrument are provided, as well as the `.wav` file of the global mix (which is a simple addition of the separated signals).

5 Data Generation

The separated tracks and the aligned MIDI scores of this dataset are created and edited as following. First, the original MIDI scores are downloaded from the Kunst der Fuge database[1] or generated from the music edition software Sibelius, and then imported in the sequencer Ableton Live.

The different tracks are then recorded separately, whilst the musicians listen to the other synthesized parts synchronized with a metronome through headphones. The recordings are afterwards edited and mixed in Digital Performer, and the MIDI scores are eventually manually aligned one by one with Sonic Visualizer [1].

We provide a visual example of a non-aligned version of the MIDI score from a clarinet and an aligned version of the same extract (see Figures 1 and 2).

6 Reference

[1] C. Cannam, C. Landone, and M. Sandler. Sonic visualiser: an open source application for viewing, analysing, and annotating music audio files. In *Proceedings of the international conference on Multimedia*, pages 1467–1468, Firenze, Italy, 2010

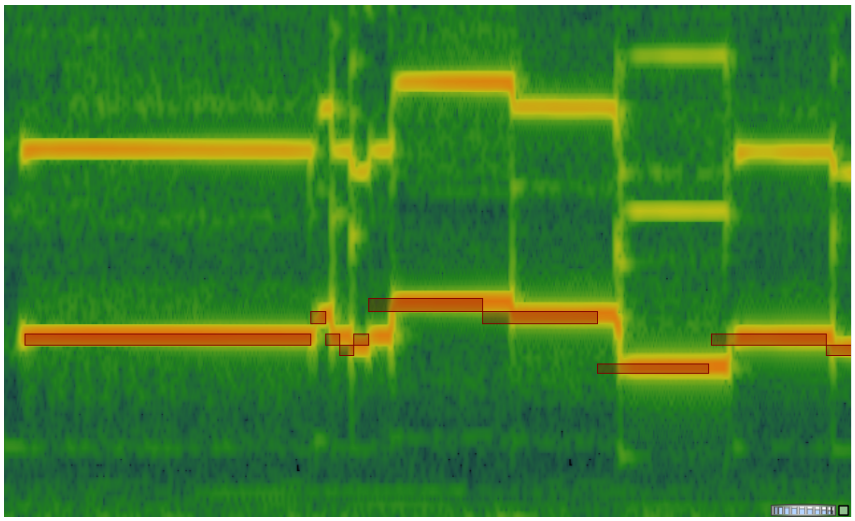


Figure 1: Non-aligned MIDI score of a clarinet extract

[1] `http://www.kunstderfuge.com/`

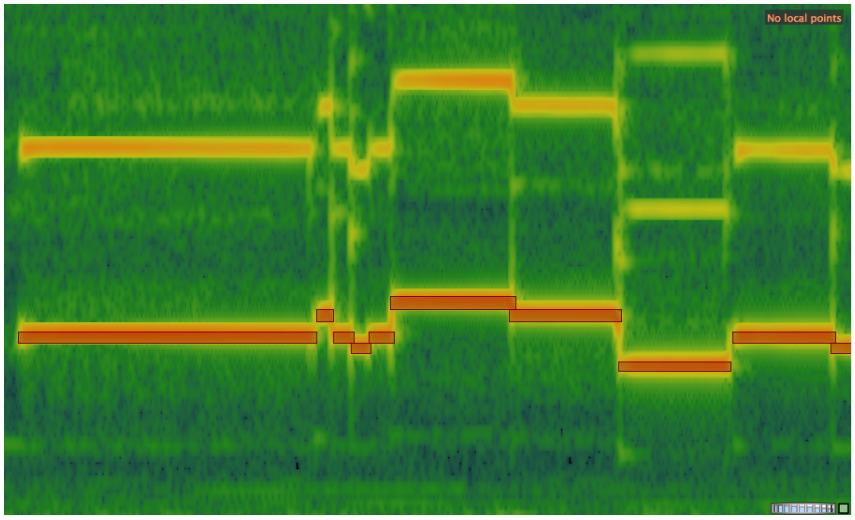


Figure 2: Aligned MIDI score of a clarinet extract