

Combining blockwise and multi-coefficient stepwise approches in a general framework for online audio source separation

Laurent S. R. Simon, Emmanuel Vincent

▶ **To cite this version:**

Laurent S. R. Simon, Emmanuel Vincent. Combining blockwise and multi-coefficient stepwise approches in a general framework for online audio source separation. [Research Report] RR-8766, Inria. 2015, pp.18. hal-01186948

HAL Id: hal-01186948
https://inria.hal.science/hal-01186948

Submitted on 25 Aug 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

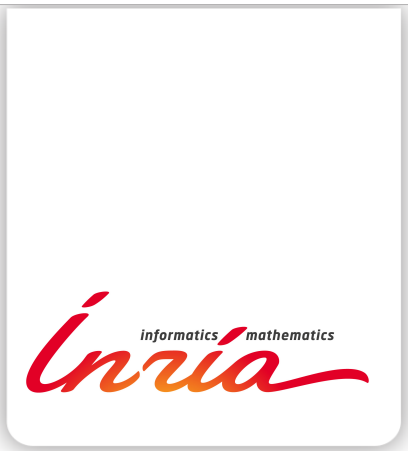


Combining blockwise and multi-coefficient stepwise approches in a general framework for online audio source separation

Laurent S. R. Simon, Emmanuel Vincent

RESEARCH REPORT
N° 8766
August 2015
Project-Team Multispeech

ISSN 0249-6399 ISRN INRIA/RR--8766--FR+ENG



Combining blockwise and multi-coefficient stepwise approches in a general framework for online audio source separation

Laurent S. R. Simon*, Emmanuel Vincent†

Project-Team Multispeech

Research Report n° 8766 — August 2015 — 20 pages

Abstract: This article considers the problem of online audio source separation. Various algorithms can be found in the literature, featuring either blockwise or stepwise approaches, and using either the spectral or spatial characteristics of the sound sources of a mixture. We offer an algorithm that can combine both stepwise and blockwise approaches, and that can use spectral and spatial information. We propose a method for pre-processing the data of each block and offer a way to deduce an Equivalent Rectangular Bandwith time-frequency representation out of a Short-Time Fourier Transform. The efficiency of our algorithm is then tested for various parameters and the effect of each of those parameters on the quality of separation and on the computation time is then discussed.

Key-words: online audio source separation, nonnegative matrix factorisation, sliding block, stochastic gradient.

* L. S. R. Simon is with LIMSI-CNRS (e-mail: laurent.simon@limsi.fr).
† E. Vincent is with Inria (e-mail: emmanuel.vincent@inria.fr).

RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Combinaison des approches par blocs et pas-à-pas en un cadre général pour la séparation en ligne de sources audio

Résumé : Cet article traite du problème de la séparation en line de sources audio. Les différents algorithmes présentés dans la littérature sont basés soit sur une approche par blocs soit sur une approche pas-à-pas et ils utilisent soit les caractéristiques spectrales soit les caractéristiques spatiales des sources sonores qui composent le mélange. Nous proposons un algorithme qui peut combiner les approches par blocs et pas-à-pas et utiliser à la fois des informations spectrales et spatiales. Nous introduisons une méthode de pré-traitement des données sur chaque bloc et une façon de calculer une représentation temps-fréquence sur l'échelle *Equivalent Rectangular Bandwith* à partir d'une transformation de Fourier à court terme. Nous évaluons la performance de notre algorithme avec différentes valeurs de paramètres et discutons l'effet de ces valeurs sur la qualité de séparation et le temps de calcul.

Mots-clés : séparation en ligne de sources audio, factorisation matricielle positive, bloc coulissant, gradient stochastique.

1 Introduction

Audio source separation is the process of recovering a set of audio signals from a given mixture signal. This can be addressed via established approaches such as Independent Component Analysis (ICA), binary masking, and Sparse Component Analysis (SCA) [1], or more recent approaches such as local Gaussian modeling and Nonnegative Matrix Factorisation (NMF) [2]. Most algorithms are *offline* (also known as batch) algorithms which process the mixture signal as a whole. In this paper, we focus on *online* audio source separation instead, whereby only the past samples of the mixture are available in order to estimate the sources at a given time. This constraint arises in particular in real-time scenarios, such as real-time speech separation for voice command, live remixing of songs by disk-jockeys, or spatial upmixing of streaming audio, as may be necessary when the 3D audio processing is performed at the consumer's side of the audio chain, like in the case in the BiLi project.

A few online implementations have been designed for time-domain ICA [3–6], frequency-domain ICA [7–12] and post-filtering [13–15], localisation-based time-frequency masking [16–21], spectral continuity-based separation [22], pitch-based separation [23] and NMF [24–27]. However, these algorithms rely either on spatial diversity [13,15,19,20] or on spectral diversity [22,24,25,27] alone. Such algorithms are not capable of separating mixtures where several sources have the same spatial position and several sources have similar spectral characteristics. For example, in pop music, the voice, the snare drum, the bass drum and the bass are often mixed to the centre and several voices or several guitars are present.

To address this issue, we adopt the FASST audio source separation framework introduced in [28]. This framework generalises a number of algorithms such as certain forms of ICA and NMF, and it enables the specification of additional constraints on the source spectra such as harmonicity. By jointly exploiting spatial and spectral diversity, it makes it possible to robustly separate difficult mixtures such as above. This is exemplified by the fact that FASST performed best among the two algorithms [28, 29] that suceeded in separating all sources from professionally mixed music recordings in the 2011 Signal Separation Evaluation Campaign (SiSEC) [30].

Approaches for online audio source separation fall into three categories. A first set of approaches operates on a single time frame of the input time-frequency representation. This has been used for localisation-based and pitch-based separation [17, 23], but this does not allow tracking of parameters such as source positions or source spectra over time. The sliding block (also known as *blockwise*) approach, as used in, e.g., [13, 15, 22, 27], consists in applying the offline audio source separation algorithm to a block of M time frames. Once the model parameters have been estimated from the observed signal in the whole block, they are used to separate the mixture in one or more time frames (e.g, the last ones of the block) before sliding the processing block by that number of frames. This approach is computationally costly but accurate because the parameters used to separate each time frame are estimated from a larger block [13]. The stochastic gradient (also known as *stepwise*) approach, as used in, e.g., [7, 16, 24, 25], offers to update the model parameters in each time frame by interpolating the parameters of the previous frame with some estimate of the parameters in the current frame weighted by a step size coef-

ficient α. This approach, which is commonly used for dictionary learning and optimisation [31–33], is faster than the blockwise approach but it can lead to less accurate parameter estimates. A few authors combined the blockwise and the stepwise approaches [11, 24] but they did not provide an experimental assessment of the benefit of this combination in terms of source separation quality.

In this paper, we propose an iterative online algorithm for the FASST framework that combines the blockwise and the stepwise approaches[1]. By contrast with the above algorithms, our algorithm is more general in that it can jointly exploit spatial and spectral diversity and it relies on three hyper-parameters: the block size M and two step sizes α_{spat} and α_{spec} which make it possible to adapt the spatial parameters and the spectral parameters at different rates. Informal tests showed an improvement of separation when using ERB filterbanks over when using Short-Time Fourier Transform (STFT); in the validation experiment, we therefore approximated an ERB filterbank by averaging over the frequency bins of a STFT. As a by-product, we provide a way of circumventing the annealing procedure in the original FASST algorithm in [28], which would require a large number of iterations per block. Finally, we assess the benefit of *preiterations*, i.e., iterations to update the temporal parameters of the last frames prior to updating all the other parameters. We assess the impact of these different hyper-parameters experimentally on a set of real-world music mixtures and show that whilst the set of parameters that achieves the best separation quality is content-dependent, one can find a set of parameters for which the separation quality is comparable to that achieved using the optimal parameters.

The structure of the rest of the paper is as follows. The original offline framework is summarized in Section 2. Section 3 presents the proposed online algorithm. Objective experiments are presented in Section 4. Conclusions and future perspectives are drawn in Section 5.

2 General audio source separation framework

2.1 Model

We operate in the time-frequency domain by means of the Short-Time Fourier Transform (STFT). In each frequency bin $f \in [1, F]$ and each time frame $n \in [1, N]$, where F is the number of time-frequency bins in a time frame and N is the total number of time frames, the multichannel mixture signal $\mathbf{x}(f, n)$ can be expressed as

$$\mathbf{x}(f, n) = \sum_{j=1}^{J} \mathbf{c}_j(f, n) \tag{1}$$

where J is the number of sources and $\mathbf{c}_j(f, n)$ is the STFT of the multichannel signal of the j-th source.

We assume that $\mathbf{c}_j(f, n)$ is a complex-valued Gaussian random vector with zero mean and covariance matrix $\mathbf{R}_{\mathbf{c}_j}(f, n)$

$$\mathbf{c}_j(f, n) \sim \mathcal{N}_c(\mathbf{0}, \mathbf{R}_{\mathbf{c}_j}(f, n)) \tag{2}$$

[1] A preliminary version of this online algorithm using a single step size α and smaller-scale experimental evaluation was introduced in [34], without using pre-iterations.

and that $\mathbf{R}_{\mathbf{c}_j}(f,n)$ factors as

$$\mathbf{R}_{\mathbf{c}_j}(f,n) = v_j(f,n)\mathbf{R}_j(f) \tag{3}$$

where $\mathbf{R}_j(f)$ is the spatial covariance matrix of the j-th source, and $v_j(f,n)$ is its spectral power.

The spectral power $v_j(f,n)$ is modeled via a form of hierarchical NMF [28], as shown in the example in Fig. 1. The matrix of spectral variances $\mathbf{V}_j \triangleq [v_j(f,n)]_{f,n}$ is first decomposed into the product of an excitation spectral power $\mathbf{V}_j^{\mathrm{x}}$ and a filter spectral power $\mathbf{V}_j^{\mathrm{f}}$

$$\mathbf{V}_j = \mathbf{V}_j^{\mathrm{x}} \odot \mathbf{V}_j^{\mathrm{f}} \tag{4}$$

where $\odot$ denotes entrywise multiplication. $\mathbf{V}_j^{\mathrm{x}}$ is further decomposed into the product of a matrix of narrowband spectral patterns $\mathbf{W}_j^{\mathrm{x}}$, a matrix of spectral envelope weights $\mathbf{U}_j^{\mathrm{x}}$, a matrix of temporal envelope weights $\mathbf{G}_j^{\mathrm{x}}$, and a matrix of time-localised temporal patterns $\mathbf{H}_j^{\mathrm{x}}$, so that

$$\mathbf{V}_j^{\mathrm{x}} = \mathbf{W}_j^{\mathrm{x}}\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}}. \tag{5}$$

$\mathbf{V}_j^{\mathrm{f}}$ is decomposed in a similar way.

This factorisation enables the specification of various spectral or temporal constraints over the sources. For example, harmonicity can be enforced by fixing $\mathbf{W}_j^{\mathrm{x}}$ to a set of narrowband harmonic patterns [28]. For other examples of use of FASST, see [35–37], or Fig. 1. This figure shows how the notes played by a guitar get decomposed into spectral information, which contains the frequencies present in each of the notes played by the guitar, and temporal information, which shows when each note is played. Fig. 1 also shows that each note is composed of several spectral patterns. In the case where the spectral information would be set, e.g. in a case where the spectral fine structures and spectral enveloppes of the guitar would have been learned prior to the separation, matrices $\mathbf{W}_j^{\mathrm{x}}$ and $\mathbf{U}_j^{\mathrm{x}}$ would be fixed while matrices $\mathbf{G}_j^{\mathrm{x}}$ and $\mathbf{H}_j^{\mathrm{x}}$ would be left uncontrained.

2.2 Offline estimator

In an offline context, the model parameters are estimated in the Maximum Likelihood (ML) sense.

The log-likelihood $\log \mathcal{L}$ is defined using the empirical mixture covariance matrix $\widehat{\mathbf{R}}_{\mathbf{x}}(f,n)$ [38] as

$$\log \mathcal{L} = \sum_{f,n} -\operatorname{tr}\left(\mathbf{R}_{\mathbf{x}}^{-1}(f,n)\widehat{\mathbf{R}}_{\mathbf{x}}(f,n)\right) - \log\det(\pi\mathbf{R}_{\mathbf{x}}(f,n)) \tag{6}$$

where

$$\mathbf{R}_{\mathbf{x}}(f,n) = \sum_{j=1}^{J} \mathbf{R}_{\mathbf{c}_j}(f,n) \tag{7}$$

is the mixture covariance predicted by the model.

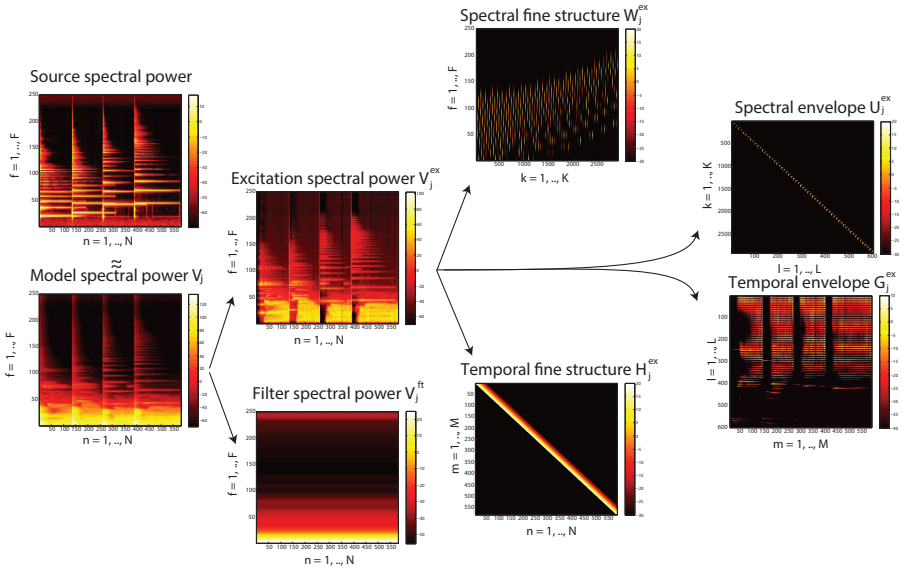


Figure 1: Example of FASST representation of a sequence of notes played by a guitar. Units are in dB.

In [28], $\mathbf{R}_j(f)$ was expressed as $\mathbf{R}_j(f) = \mathbf{A}_j(f)\mathbf{A}_j^H(f)$, and $\mathbf{A}_j(f)$ was estimated instead. This resulted in an annealing procedure, which would translate into a large number of iterations per block in our context. In order to circumvent it, we assume that $\mathbf{R}_j(f)$ is full-rank and then we directly estimate $\mathbf{R}_j(f)$ instead, similarly to [38].

A Generalised Expectation-Maximisation (GEM) algorithm combined with Multiplicative Updates (MU) is then applied to the complete data $\{\mathbf{c}_j(f,n)\}$. In the E-step, the natural statistics are computed as per [38]

$$\mathbf{\Omega}_j(f,n) = \mathbf{R}_{\mathbf{c}_j}(f,n)\mathbf{R}_{\mathbf{x}}^{-\mathbf{1}}(f,n) \tag{8}$$

$$\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n) = \mathbf{\Omega}_j(f,n)\widehat{\mathbf{R}}_{\mathbf{x}}(f,n)\mathbf{\Omega}_j^H(f,n) + (\mathbf{I} - \mathbf{\Omega}_j(f,n))\mathbf{R}_{\mathbf{c}_j}(f,n) \tag{9}$$

where $\mathbf{\Omega}_j$ is the multichannel Wiener filter, $\mathbf{I}$ is the $I \times I$ identity matrix and I is the number of channels of the mixture.

In the M-step, the model parameters are updated as in [28,38] by maximizing the cost function Q in (33), which leads by derivation of Q to the updates shown in (11), (12), (13), (14) and (15).

$$Q = \sum_{j,f,n} -\operatorname{tr}\left(\mathbf{R}_{\mathbf{c}_j}^{-1}(f,n)\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)\right) - \log\det\mathbf{R}_{\mathbf{c}_j}(f,n) \tag{10}$$

$$\mathbf{R}_j(f) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{v_j(f,n)}\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n) \tag{11}$$

$$\mathbf{W}_j^{\mathrm{x}} = \mathbf{W}_j^{\mathrm{x}} \odot \frac{[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}](\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T}{\mathbf{V}_j^{\mathrm{x}}.^{-1}(\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T} \tag{12}$$

$$\mathbf{U}_j^{\mathrm{x}} = \mathbf{U}_j^{\mathrm{x}} \odot \frac{\mathbf{W}_j^{\mathrm{x}T}[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}](\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T}{\mathbf{W}_j^{\mathrm{x}T}\mathbf{V}_j^{\mathrm{x}}.^{-1}(\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T} \tag{13}$$

$$\mathbf{G}_j^{\mathrm{x}} = \mathbf{G}_j^{\mathrm{x}} \odot \frac{(\mathbf{W}_j^{\mathrm{x}}\mathbf{U}_j^{\mathrm{x}})^T[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}]\mathbf{H}_j^{\mathrm{x}T}}{(\mathbf{W}_j^{\mathrm{x}}\mathbf{U}_j^{\mathrm{x}})^T\mathbf{V}_j^{\mathrm{x}}.^{-1}\mathbf{H}_j^{\mathrm{x}T}} \tag{14}$$

$$\mathbf{H}_j^{\mathrm{x}} = \mathbf{H}_j^{\mathrm{x}} \odot \frac{(\mathbf{W}_j^{\mathrm{x}}\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}})^T[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}]}{(\mathbf{W}_j^{\mathrm{x}}\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}})^T\mathbf{V}_j^{\mathrm{x}}.^{-1}} \tag{15}$$

In these equations, $.^p$ denotes entrywise raising to the power p, N is the number of time frames in the STFT of the signal, and $\widehat{\mathbf{\Xi}}_j = [\widehat{\xi}_j(f,n)]_{f,n}$, where $\widehat{\xi}_j(f,n)$ is given by

$$\widehat{\xi}_j(f,n) = \frac{1}{I}\operatorname{tr}(\mathbf{R}_j^{-1}(f)\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)). \tag{16}$$

$\mathbf{W}_j^{\mathrm{f}}$, $\mathbf{U}_j^{\mathrm{f}}$, $\mathbf{G}_j^{\mathrm{f}}$ and $\mathbf{H}_j^{\mathrm{f}}$ are updated in a similar way.
The separated sources are then obtained via multichannel Wiener filtering as

$$\widehat{\mathbf{c}}_j(f,n) = \mathbf{\Omega}_j(f,n)\mathbf{x}(f,n). \tag{17}$$

3 Online estimation

3.1 Combined blockwise and stepwise approach

We now consider an online estimation context. We split the data into blocks indexed by t and shifted by D STFT frames. Each block covers M STFT frames indexed by n with $tD-M+1 \leq n \leq tD$, where $M = 1$ for the stepwise approach and $M = N$ for the full offline approach. Figure 2 illustrates the decomposition of $\mathbf{x}(f,n)$ into blocks.

At each block, several iterations can be performed in order to estimate the model parameters. At each iteration, the expectation of the natural statistics is computed using (8) and (9) for $tD - M + 1 \leq n \leq tD$.

The algorithm in Sec. 2 is therefore adapted to the online constraint. The E-step remains the same as for the offline approach, applied to the blocks of M frames. The M step is modified by applying a weighted averaging over time. The original Q function in (33) is therefore modified into

$$\begin{aligned} Q^{(t)} &= (1-\alpha)Q^{(t-1)} \\ &\quad + \alpha\left(\sum_{j,f}\sum_{n=tD-M+1}^{tD} - \operatorname{tr}\left(\mathbf{R}_{\mathbf{c}_j}^{-1}(f,n)\widehat{\mathbf{R}}_{\mathbf{c}_j}^{(}f,n\right) - \log\det\mathbf{R}_{\mathbf{c}_j}(f,n)\right) \end{aligned} \tag{18}$$

where α is a step size coefficient $\alpha \in]0;1]$ used to stabilise the parameter updates by averaging over blocks.

We assume that the ideal spatial and spectral parameters vary little from one block to the next. The partial derivatives of (18) leads to the new updates of $\mathbf{R}_j^{(t)}$, $\mathbf{U}_j^{\mathrm{x}(t)}$ and $\mathbf{W}_j^{\mathrm{x}(t)}$ for the block of signal available at time t.

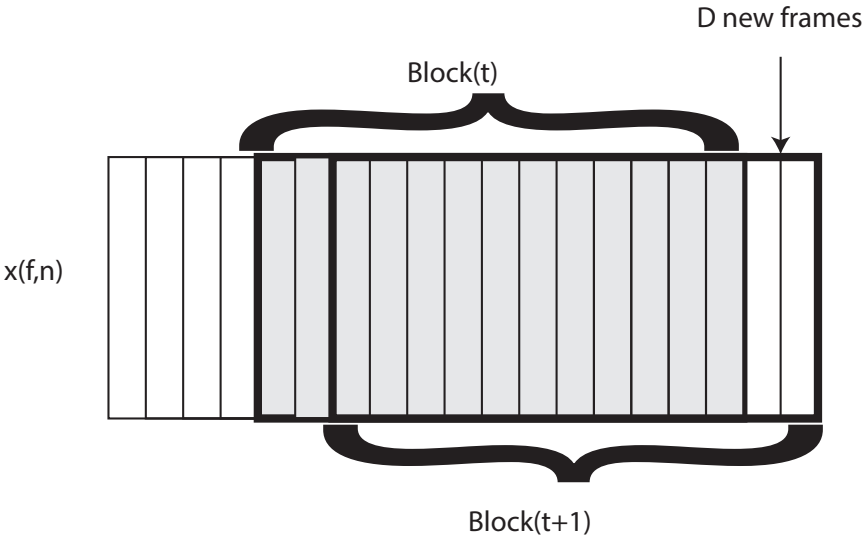


Figure 2: Evolution of the blocks of signal across time.

The spatial covariance matrix is then updated as follows:

$$\mathbf{R}_j^{(t)}(f) = (1-\alpha)\mathbf{R}_j^{(t-1)}(f) + \alpha\left(\frac{1}{M}\sum_{n=tD-M+1}^{tD}\frac{1}{v_j(f,n)}\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)\right) \quad (19)$$

where the superscript $^{(t)}$ denotes the estimated parameters for the block t.

The temporal envelope weights $\mathbf{G}_j^{\mathrm{x}(t)}$ and the time-localised temporal patterns $\mathbf{H}_j^{\mathrm{x}(t)}$ are updated using (14) and (15) for $tD - M + 1 \leq n \leq tD$, as they are expected to significantly vary between blocks.

The variations of the narrowband spectral patterns $\mathbf{W}_j^{\mathrm{x}(t)}$ and of the spectral envelope weights $\mathbf{U}_j^{\mathrm{x}(t)}$ are expected to be smaller and can be smoothed using the stepwise approach. The updates are therefore calculated by using the derivative of (18) in reference to $\mathbf{W}_j^{\mathrm{x}}$ given in (20) and that of (18) in reference to $\mathbf{U}_j^{\mathrm{x}}$ given in (21). This leads to the updates of $\mathbf{W}_j^{\mathrm{x}(t)}$ and of $\mathbf{U}_j^{\mathrm{x}(t)}$ in (26) and (27). Detailed derivation is given in Sec. A.
where $\widehat{\mathbf{\Xi}}_j^{(t)}$ is computed as in (16). $\mathbf{G}_j^{\mathrm{f}\,(t)}$, $\mathbf{H}_j^{\mathrm{f}\,(t)}$, $\mathbf{W}_j^{\mathrm{f}\,(t)}$, and $\mathbf{U}_j^{\mathrm{f}\,(t)}$ are updated in a similar way.

After each EM iteration, $\mathbf{W}_j^{\mathrm{f}}$, $\mathbf{U}_j^{\mathrm{f}}$, $\mathbf{G}_j^{\mathrm{f}}$ and $\mathbf{H}_j^{\mathrm{f}}$, $\mathbf{W}_j^{\mathrm{x}}$, $\mathbf{U}_j^{\mathrm{x}}$, and $\mathbf{G}_j^{\mathrm{x}}$ are normalized to 1 and $\mathbf{H}_j^{\mathrm{x}}$ is normalized to the energy of $\mathbf{V}\dot{\mathrm{j}}$.

Although (19) to (27) look similar to the stepwise local Gaussian model update in [39] and the stepwise NMF updates in [25], there are two crucial differences:

- The framework introduced in the current paper is more general in the sense that it uses hierarchical NMF, enabling the user to apply more specific constraints than when using shallow NMF and to solve more difficult separation problems sur as such as those arising in music.
- The models it estimates take into account both the source's spatial and spectral diversity.

$$\frac{\partial Q^{(t)}}{\partial \mathbf{W}_j^{\mathrm{x}}} = \mathbf{M}_j^{\mathrm{x}(t)} - \mathbf{C}_j^{\mathrm{x}(t)} \tag{20}$$

$$\frac{\partial Q^{(t)}}{\partial \mathbf{U}_j^{\mathrm{x}}} = \mathbf{N}_j^{\mathrm{x}(t)} - \mathbf{D}_j^{\mathrm{x}(t)} \tag{21}$$

where

$$\mathbf{M}_j^{\mathrm{x}(t)} = (1-\alpha)\mathbf{M}_j^{\mathrm{x}(t-1)} + \alpha[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}](\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \tag{22}$$

$$\mathbf{C}_j^{\mathrm{x}(t)} = (1-\alpha)\mathbf{C}_j^{\mathrm{x}(t-1)} + \alpha\mathbf{V}_j^{\mathrm{x}}.^{-1}(\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \tag{23}$$

$$\mathbf{N}_j^{\mathrm{x}(t)} = (1-\alpha)\mathbf{N}_j^{\mathrm{x}(t-1)} + \alpha\mathbf{W}_j^{\mathrm{x}T}[\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1}](\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \tag{24}$$

$$\mathbf{D}_j^{\mathrm{x}(t)} = (1-\alpha)\mathbf{D}_j^{\mathrm{x}(t-1)} + \alpha\mathbf{W}_j^{\mathrm{x}T}\mathbf{V}_j^{\mathrm{x}}.^{-1}(\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \tag{25}$$

$$\mathbf{W}_j^{\mathrm{x}(t)} = \mathbf{W}_j^{\mathrm{x}(t)} \odot \frac{\mathbf{M}_j^{\mathrm{x}(t)}}{\mathbf{C}_j^{\mathrm{x}(t)}} \tag{26}$$

$$\mathbf{U}_j^{\mathrm{x}(t)} = \mathbf{U}_j^{\mathrm{x}(t)} \odot \frac{\mathbf{N}_j^{\mathrm{x}(t)}}{\mathbf{D}_j^{\mathrm{x}(t)}}. \tag{27}$$

- It is not limited to the sole use of the latest audio frame: the algorithm blockwise in addition to being stepwise.

3.2 Different α coefficients for spatial and spectral parameters

We found in [34] that the use of a single stepsize coefficient α tended to cause a divergence of the spatial parameters over time and therefore a degradation of the separation quality. In addition, in most commercial music recordings — the intended application context of this article —, the spatial position of the sources evolves slowly if at all, contrarily to their spectro-temporal content. In other source separation contexts, we may aim to separate moving sources that change little in terms of spectral content.

We therefore propose in this paper to use two distinct stepsize coefficients in $]0;1]$: a spatial stepsize coefficient α_{spat} in (19) and a spectral stepsize coefficient α_{spec} in (22) to (25).

3.3 Initialisation and preiterations

The spatial covariance matrices $\mathbf{R}_j^{(t)}$ are initialised to $\mathbf{R}_j^{(t-1)}$. The narrowband spectral patterns $\mathbf{W}_j^{\mathrm{x}(t)}$, the spectral envelope weights $\mathbf{U}_j^{\mathrm{x}(t)}$, and the temporal

envelope weights $\mathbf{G}_j^{\mathrm{x}(t)}$ are initialised to $\mathbf{W}_j^{\mathrm{x}(t-1)}$,$\mathbf{U}_j^{\mathrm{x}(t-1)}$, and $\mathbf{G}_j^{\mathrm{x}(t-1)}$.

The temporal weights of the last D frames of $\mathbf{H}_j^{\mathrm{x}(t)}$ are randomly initialised and then normalised to the mean spectral power of the signal:

$$\mathbf{H}_j^{\mathrm{x}(t)}(p,k) = \epsilon_j(p,k,t)\tilde{\mathbf{V}}_j \tag{28}$$

where $\tilde{\mathbf{V}}_j$ is defined by

$$\tilde{\mathbf{V}}_j = \frac{1}{FM} \sum_{f,n=tD-M+1}^{tD} \widehat{\mathbf{\Xi}}_j^{(t)} \tag{29}$$

and $\epsilon_j(p,k,t)$ is defined by

$$\epsilon_j(p,k,t) = 0.75 \mid \kappa \mid +0.5 \tag{30}$$

and κ is a random variable following a normal distribution.

The remaining of $\mathbf{H}_j^{\mathrm{x}(t)}$ is initialised to

$$\mathbf{H}_j^{\mathrm{x}(t)}(p,k) = \mathbf{H}_j^{\mathrm{x}(t-1)}(p,k+D) + \gamma\epsilon_j(p,k,t)\tilde{\mathbf{V}}_j \tag{31}$$

for all p and $k \in [1; M-D]$. The role of the added noise is to prevent the estimator from being locked to a zero value if there was a zero in $\mathbf{H}_j^{\mathrm{x}(t-1)}$.

$\mathbf{W}_j^{\mathrm{f}(t)}$, $\mathbf{U}_j^{\mathrm{f}(t)}$, $\mathbf{G}_j^{\mathrm{f}(t)}$, and $\mathbf{H}_j^{\mathrm{t}(t)}$ are initialized in a similar way to $\mathbf{W}_j^{\mathrm{x}(t)}$, $\mathbf{U}_j^{\mathrm{x}(t)}$, $\mathbf{G}_j^{\mathrm{x}(t)}$, and $\mathbf{H}_j^{\mathrm{x}(t)}$.

Running the algorithm described above on each new block may have a drawback: since the temporal weights of the new D frames are randomly initialised, the source models may start to converge for the first few iterations of the $M-D$ first frames of the block towards a local optimum that would lead to a bad separation. To avoid that, we propose to preprocess the last frame of each block by running the algorithm on the whole block while fixing all the parameters of the models except the temporal weights of the last D frames. An iteration of this preprocessing is what we call a preiteration.

4 Experimental evaluation

4.1 ERB filterbank approximation from an STFT

Online processing of filterbanks is not a trivial matter. However, as explained in Section 1, ERB filterbanks lead to a better source separation than STFT representations of the signal. For this reason, we chose to simulate an ERB filterbank by concatenating for each time frame different bins of the covariance matrix of the mixture.

Once the filters for the different sources have been estimated using the algorithm, the filters are applied by expanding the ERB representation of the filters into a STFT representation of the filters: a single ERB filter coefficient is then applied to several STFT bins of the mixture signal, using the reverse of the weighting used for the concatenation.

In equations (7), (8), (9), (16), (17), (19) and (29), we replace the frequency index f by b, where b is the index of the frequency band.

Parameter	Values
Number of iterations	[2, 5, 10, 30]
Number of pre-iterations	[0, 2, 5, 10]
M	[1, 2, 10, 25, 100]
γ	[0, 0.1, 0.25, 0.5]
α_{spat}	[0.02, 0.05, 0.1, 1]
α_{spec}	[0.02, 0.05, 0.1, 1]

Table 1: Values tested for each parameter of the proposed algorithm

4.2 Data and algorithm settings

For our experiments, we processed five 10 s long stereo commercial pop recordings taken from the QUASI database [40,41] . Each recording involves 4 sources sampled at 44.1 kHz: bass, drums, a guitar and a voice, and their respective effects (delays, reverberation, ...).

A 350 bands simulated ERB filterbank was used, following the method described in Section 4.1. The energy in the ERB bands was estimated using an averaging over the frequency bins of a 2048 point STFT. For the offline algorithm as well as for the online algorithm, each of the modeled sources were constrained in a way similar to Section V.C in [28]. In the case of an harmonic source, $\mathbf{W}_j^{\mathrm{x}(t)}$ was fixed to a set of narrowband harmonic spectral patterns and the spectral envelope weights in $\mathbf{U}_j^{\mathrm{x}(t)}$ were updated, whereas for bass and percussive sources, $\mathbf{W}_j^{\mathrm{x}(t)}$ was a fixed diagonal matrix and $\mathbf{U}_j^{\mathrm{x}(t)}$ was a fixed matrix of basis spectra learned over a corpus of bass and drum sounds.

Separation performance was estimated for 4096 different sets of conditions of the online algorithm (number of iterations, number of pre-iterations, block size m, γ, α_{spat} and α_{spec}) and 100 iterations for the offline algorithm in [28]. Table 1 specifies the values tested for each parameter. We set $D = 1$, which corresponds to a low-latency scenario.

The performance of the algorithms was evaluated with respect to the Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR), source Image to Spatial distortion Ratio (ISR) and Source-to-Artifacts Ratio (SAR) as defined in [42]. For each set of conditions, each of these criteria was averaged over all the mixtures and all the separated sound sources.

The conditions where the block size M equals 1 are equivalent to a pure stepwise approach, while the conditions where α_{spat} and α_{spec} equal 1 are equivalent to a pure blockwise approach.

4.3 Results per song

Table 2 reports the SDR measured for each song's best conditions, SDR measured for the overall best condition, SDR measured offline and for the best blockwise and best stepwise conditions. Table 3 shows the values of the parameters that led to the online result. It can be seen that there is no consensus for the best case scenario, except for α_{spec}, where the highest value seems best for four of the five songs. For these four songs, the results obtained with the online algorithm are better than with the offline algorithm. These values may seem

Song	online individual best case SDR	online global best case SDR	Offline SDR	blockwise best case SDR	stepwise best case SDR
1	1.10	0.65	0.71	0.97	0.97
2	1.76	1.56	2.07	1.76	1.38
3	1.13	0.85	0.66	1.01	0.95
4	3.92	2.14	5.60	2.42	3.92
5	2.49	2.34	1.24	2.38	2.49

Table 2: Best case scenario online, average best case scenario online and offline SDR for each song

Song	N. iter	N. preiter	M	γ	α_{spat}	α_{spec}
1	30	2	10	0.25	1	0.02
2	10	10	100	0	0.1	1
3	2	0	10	0.1	0.05	0.1
4	5	0	2	0.5	0.1	0.1
5	30	0	2	0.1	0.05	0.02
Global best case	2	5	25	0	1	1

Table 3: Best sets of parameters for each song and on average.

Song	pure spectral SDR	pure spatial SDR	Mixed
1	0.72	0.92	0.71
2	2.08	1.24	2.07
3	0.65	0.68	0.66
4	5.73	1.79	5.60
5	1.18	1.00	1.24

Table 4: SDR for each song for the pure spectral, pure spatial and combined spectral and spatial offline algorithms.

low, but they are averaged for all sources. With the source material used in the experiment, the snare drum tended to play at the same position and at the same time as the voice, the algorithm therefore tended to associate some drum elements to the voice, thus reducing the SDR of the voice. Efficient methods for separating the voice from drums exist but are not the focus of this article.

Table 4 shows the comparison of pure spectral, pure spatial, and combined spectral and spatial algorithms for the offline framework on the five songs tested in the experiment.

4.4 Best results on average

Averaging the results over all songs gives a set of parameters that give, on average on all five extracts songs, the best SDR. Table 2 shows the SDR of each song with this set of parameters. These parameters are 2 iterations, 5 preiterations, $\gamma = 0$, $M = 25$, $\alpha_{\text{spat}} = 1$ and $\alpha_{\text{spec}} = 1$.

The parameter values for each song's best case scenario differ from one an-

Variables	d.f.	F	p
Song number*Number of iterations	12	175	0
Song number*Number of pre-iterations	12	3.7	0
Song number*M	16	79	0
Song number*γ	12	21	0
Song number*α_{spat}	12	2	0.01
Song number*α_{spec}	12	31	0

Table 5: Results of the N-way ANOVA conducted on the raw results, showing the effect of the interaction between the choice of the song and the other parameters

other, and the values of the average best set of parameters differ from all of them, making the best over all sets of parameters a blockwise condition. However, none of the individual best cases followed blockwise conditions, implying that future work should be conducted on how to estimate the best set of parameters for a given song.

Table 2 also shows the results for pure blockwise (the best case for each song with the limitation $\alpha_{\text{spat}} = \alpha_{\text{spec}} = 1$) and pure stepwise approaches ($M = 1$, $\alpha_{\text{spat}} < 1$ and $\alpha_{\text{spec}} < 1$). Our combined approach offers on average a better SDR than pure blockwise and stepwise approaches, although some individual best cases were stepwise. For the stepwise approach, the number of preiterations was set to 0, as it would act on the whole block.

4.5 Analysis of variance

In order to estimate the effect of each of the parameters on the SDR, an N-way analysis of variance (ANOVA) was performed on the results. It showed that when optimised for each song (i.e looking at the interaction between the choice of the song and each parameter), all parameters have a significant effect on the SDR ($p < 0.05$). This means that changing the value of only one of these parameters could significantly alter the SDR. However, as can be seen in Table 5, the number of pre-iterations and the value of α_{spat} have a smaller influence on the SDR than the other parameters (their F-value, indicating how strong the effect is, is smaller than for the other parameters).

Performing an ANOVA on the SDR averaged over all songs confirms the results of the earlier ANOVA and indicates that the effect of M is not as significant as the other parameters, see Table 6.

4.6 Computational power

While the F-values of the ANOVA showed us how strong was the effect of each parameter on the SDR, it tells nothing of the effect of a parameter on the computational complexity of the algorithm. As shown in Sec. 4.5, if one wants to decrease the computational power, one can do it by decreasing the number of pre-iterations without significantly decreasing the SDR.

When fixing all but one parameter, the effect of this parameter on the computational power depends on which parameter is varied. For example, unless

Variables	d.f.	F	p
Number of iterations	3	2260	0
Number of pre-iterations	3	2	0.04
M	4	24	0
γ	3	184	0
α_{spat}	3	8	0
α_{spec}	3	2101	0

Table 6: Results of the N-way ANOVA conducted on the results averaged over all songs

they are fixed to 1 (i.e a blockwise approach), α_{spat} and α_{spec} have no influence on the number of operations performed by the algorithm. Similarly, γ has no influence on the computational power unless it is set to 0, since when it is set to zero, equation (32) becomes

$$\mathbf{H}_j(p,k)^{\mathrm{x}(t)} = \mathbf{H}_j(p,k+D)^{\mathrm{x}(t-1)}. \tag{32}$$

However, even in that case, its influence is extremely small.

In order to estimate the effect of the number of iterations, the number of pre-iterations and M on the computational power, we measured the CPU time necessary to run the algorithm on the first mixture with various numbers of iterations (2, 4, 6, 8, and 10), numbers of pre-iterations (0, 2, 4, 6, 8, and 10) and values of M (2, 4, 6, and 8). The effect of one of these three parameters on the computational power was estimated by averaging the CPU time for all the other parameters.

Figure 3 shows the effect of these three parameters on the computation time (forcing the algorithm to use a single thread). It can be seen that the effect of the number of iterations is far superior to the effect of the number of pre-iterations or the size of the block M. In the algorithm, M only has an effect on the size of the matrices $\mathbf{H}_j^{\mathrm{x}(t)}$, $\mathbf{H}_j^{\mathrm{f}\,(t)}$, $\mathbf{V}_j^{\mathrm{x}(t)}$, and $\mathbf{V}_j^{\mathrm{f}\,(t)}$, whereas the number of iterations has an effect on all the operations. The effect of the number of pre-iterations is somewhat between both.

The computation time shown here makes it not possible to process the data in real-tim, but it is similar to that of the matlab implementation of the FASST framework it was based on. Version 2.0 of the FASST framework has been greatly optimized, however, and the C++ version now runs almost real time if less than 20 iterations are conducted. A similar optimization of this online algorithm will be considered as well.

5 Conclusion

In this paper, new approaches to online source separation were proposed, combining stepwise and blockwise approaches, using pre-iterations to limit divergence of the model and to speed-up the processing, and making use of two separate stepwise coefficient to adjust separately the speed of convergence of the

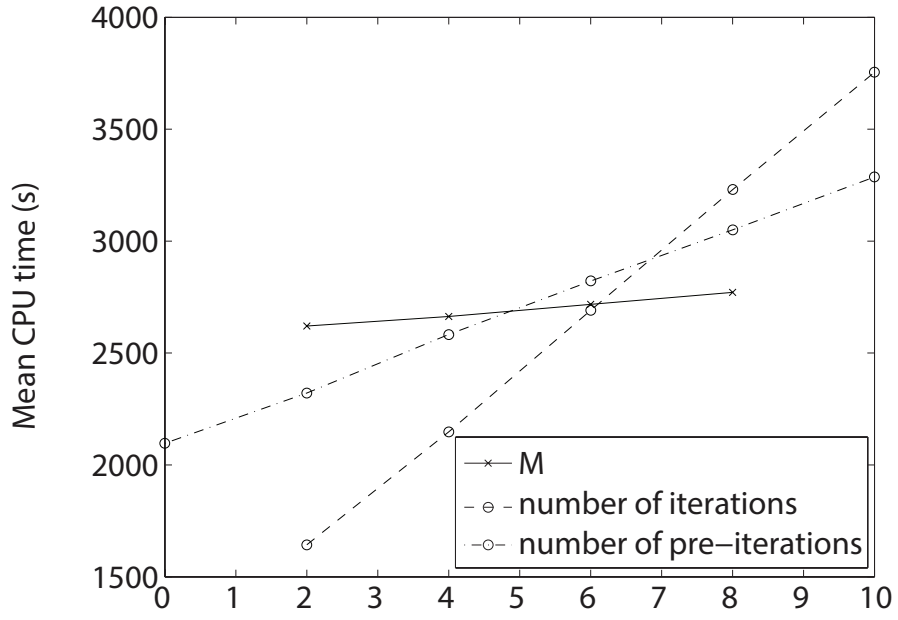


Figure 3: CPU time necessary to run the separation algorithm on a 20 seconds song as a function of number of iterations, number of pre-iterations and M.

spatial and spectral models of the sources. The use of separate stepsize coefficients for spatial and spectral parameters offers an increased flexibility in terms of constraints applicable to the source models as well as computational power. However, the ANOVA performed on the results of our experiment showed that the most computationnaly intensive parameter is also the parameter that has the most effect on the selected quality metrics of the separated audio sources.

This paper showed that the use of preiterations improved the quality of the separation at a minimal additional computational cost.

It also offered a solution to make use of ERB time-frequency representation when only STFT frames are available.

However, results of informal listening tests suggest that future work should be conducted on modifying the quality metrics (SDR, SIR, ISR, and SAR) and perceptual estimations of the PEASS toolbox to adapt it to the temporal evolutions of the models that can occur in online source separation.

The best sets of parameters were found for each song for SDR measurements and it was shown that the algorithm described in this paper perfoms on average better than the state of the art offline, as well as pure blockwise or pure stepwise blind source separation in a majority of cases for commercial recordings.

6 Acknowledgements

This work was supported by the EUREKA Eurostars i3DMusic project funded by Oseo. Additional funding was provided by the French project BiLi ("Binaural

Listening" www.bili-project.org, FUI-AAP14)

A Derivative of the Q-function

$$Q^{(t)} = (1-\alpha)Q^{(t-1)} + \alpha \left(\sum_{j,f,n} - \operatorname{tr}\left(\mathbf{R}_{\mathbf{c}_j}^{-1}(f,n)\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)\right) - \log\det \mathbf{R}_{\mathbf{c}_j}(f,n) \right) \tag{33}$$

devient

$$\begin{aligned} Q^{(t)} = (1-\alpha)Q^{(t-1)} \\ + \alpha \left(\sum_{j,f,n} - \operatorname{tr}\left(\mathbf{R}_{j}^{-1}(f)v_j^{-1}(f,n)\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)\right) - \log\det \mathbf{R}_{\mathbf{c}_j}(f,n) \right) \end{aligned} \tag{34}$$

$$Q^{(t)} = (1-\alpha)Q^{(t-1)} + \alpha \left(\sum_{j,f,n} -\frac{I\widehat{\xi}_j(f,n)}{v_j(f,n)} - I\log v_j(f,n) - \log\det \mathbf{R}_j(f) \right) \tag{35}$$

$$\frac{\partial Q^{(t)}}{\partial v_j(f,n)} = (1-\alpha)\frac{\partial Q^{(t-1)}}{\partial v_j(f,n)} + \alpha \left(\frac{I\widehat{\xi}_j(f,n)}{v_j^2(f,n)} - \frac{I}{v_j(f,n)} \right), \tag{36}$$

therefore

$$\frac{\partial Q^{(t)}}{\partial \mathbf{V}_j} = (1-\alpha)\frac{\partial Q^{(t-1)}}{\partial \mathbf{V}_j} + \alpha \left(I\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j.^{-2} - I\mathbf{V}_j.^{-1} \right). \tag{37}$$

Hence, when using

$$\frac{\partial Q^{(t)}}{\partial \mathbf{W}_j^{\mathrm{x}}} = \frac{\partial Q^{(t-1)}}{\partial \mathbf{V}_j}\frac{\partial \mathbf{V}_j}{\partial \mathbf{V}_j^{\mathrm{x}}}\frac{\partial \mathbf{V}_j^{\mathrm{x}}}{\partial \mathbf{W}_j^{\mathrm{x}}}, \tag{38}$$

we obtain

$$\begin{aligned} \frac{\partial Q^{(t)}}{\partial \mathbf{W}_j^{\mathrm{x}}} = (1-\alpha)\frac{\partial Q^{(t-1)}}{\partial \mathbf{W}_j^{\mathrm{x}}} + \alpha \Big((I\widehat{\mathbf{\Xi}}_j \odot \mathbf{V}_j^{\mathrm{x}}.^{-2} \odot \mathbf{V}_j^{\mathrm{f}}.^{-1})(\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \\ - I\mathbf{V}_j.^{-1}(\mathbf{U}_j^{\mathrm{x}}\mathbf{G}_j^{\mathrm{x}}\mathbf{H}_j^{\mathrm{x}})^T \Big). \end{aligned} \tag{39}$$

According to [43], the multiplicative update is therefore obtained by multiplying with the positive component of equation 39 and dividing by its negative component, thus leading to equation 26.

References

[1] S. Makino, T.-W. Lee, and H. Sawada, *Blind Speech Separation*, 1st ed. Springer, Sep. 2007.

[2] E. Vincent, M. G. Jafari, S. A. Abdallah, M. D. Plumbley, and M. E. Davies, "Probabilistic modeling paradigms for audio source separation," in *Machine Audition: Principles, Algorithms and Systems*. IGI Global, 2010, pp. 162–185. [Online]. Available: http://hal.inria.fr/inria-00544016/en/

[3] H. Buchner, R. Aichner, and W. Kellermann, "A generalization of blind source separation algorithms for convolutive mixtures based on second-order statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 120–134, 2005.

[4] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, "A real-time blind source separation scheme and its application to reverberant and noisy acoustic environments," *Signal Processing*, vol. 86, pp. 1260–1277, 2006.

[5] P. B. Batalheiro, M. R. Petraglia, and D. B. Haddad, "Online subband blind source separation for convolutive mixtures using a uniform filter bank with critical sampling," in *Proceedings of the 8th International Conference on Independent Component Analysis and Signal Separation*, 2009, pp. 211–218. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-00599-2˙27

[6] J. Málek, Z. Koldovský, and P. Tichavský, "Adaptive time-domain blind separation of speech signals," in *Proceedings of the 9th International Conference on Latent Variable Analysis and Signal Separation*, 2010, pp. 9–16. [Online]. Available: http://dl.acm.org/citation.cfm?id=1929142.1929145

[7] N. Murata and S. Ikeda, "An on-line algorithm for blind source separation on speech signals." [Online]. Available: http://www.academia.edu/2875529/An˙on-line˙algorithm˙for˙blind˙source˙separation˙on˙speech˙signals

[8] J. Anemüller and T. Gramss, "On-line blind separation of moving sound sources," in *Proceedings of the 1st International Workshop on Independent Component Analysis and Blind Signal Separation*, 1999, pp. 331–334.

[9] L. Parra and C. Spence, "On-line convolutive blind source separation of non-stationary signals," *Journal of VLSI Signal Processing Systems*, vol. 26, no. 1/2, pp. 39–46, Aug. 2000. [Online]. Available: http://dx.doi.org/10.1023/A:1008187132177

[10] M. S. Pedersen, U. Kjems, K. B. Rasmussen, and L. K. Hansen, "Semi-blind source separation using head-related transfer functions," in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, 2004, pp. V–713–716.

[11] N. Q. K. Duong, C. Park, and S.-H. Nam, "Application of block-online blind source separation to acoustic echo cancellation," *Journal of the Acoustical Society of Korea*, vol. 27, no. 1, pp. 17–24, 2008.

[12] F. Nesta, T. Wada, and B.-H. Juang, "Batch-online semi-blind source separation applied to multi-channel acoustic echo cancellation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 583–599, 2011.

[13] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Robust real-time blind source separation for moving speakers in a room," in *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, 2003, pp. V–469–472.

[14] J.-M. Valin, J. Rouat, and F. Michaud, "Enhanced robot audition based on microphone array source separation with post-filter," in *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, 2004, pp. 2123–2128.

[15] Y. Mori, H. Saruwatari, T. Takatani, S. Ukai, K. Shikano, T. Hiekata, Y. Ikeda, H. Hashimoto, and T. Morita, "Blind separation of acoustic signals combining SIMO-model-based independent component analysis and binary masking," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, 2006, article ID 34970. [Online]. Available: http://www.hindawi.com/journals/asp/2006/034970/abs/

[16] S. Rickard, R. Balan, and J. Rosca, "Real-time time-frequency based blind source separation," in *in Proc. of International Conference on Independent Component Analysis and Signal Separation (ICA2001*, 2001, pp. 651–656.

[17] D. Barry, R. Lawlor, and E. Coyle, "Real-time sound source separation: Azimuth discrimination and resynthesis," in *Proceedings of the 117th AES Convention*, 2004, paper Number 6258. [Online]. Available: http://arrow.dit.ie/argcon/35

[18] N. Mitianoudis and T. Stathaki, "Batch and online underdetermined source separation using Laplacian mixture models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1818–1832, 2007.

[19] B. Loesch and B. Yang, "Online blind source separation based on time-frequency sparseness," in *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2009, pp. 117–120. [Online]. Available: http://dl.acm.org/citation.cfm?id=1584263

[20] P. Pertilä, "Online blind speech separation using multiple acoustic speaker tracking and time-frequency masking," *Computer Speech and Language*, vol. 27, no. 3, pp. 683–702, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0885230812000630

[21] M. Souden, K. Kinoshita, and T. Nakatani, "Towards online maximum-likelihood-based speech clustering and separation," *Journal of the Acoustical Society of America*, vol. 133, no. 5, pp. EL339–345, 2013.

[22] N. Ono, K. Miyamoto, and S. Sagayama, "A real-time equalizer of harmonic and percussive components in music signals," in *Proceedings of the 9th International Conference on Music Information Retrieval*, 2008, pp. 139–144.

[23] R. Marxer and J. Janer, "A Tikhonov regularization method for spectrum decomposition in low latency audio source separation," in *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2012, pp. 277–280.

[24] A. Lefèvre, F. Bach, and C. Févotte, "Online algorithms for nonnegative matrix factorization with the Itakura-Saito divergence," in *Proceedings of the 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2011, arXiv e-print, pp. 313–316. [Online]. Available: http://arxiv.org/abs/1106.4198

[25] D. Wang, R. Vipperla, and N. Evans, "Online pattern learning for non-negative convolutive sparse coding," in *Proceedings of Interspeech*, 2011, pp. 65–68.

[26] Z. Duan, G. J. Mysore, and P. Smaragdis, "Online PLCA for real-time semi-supervised source separation," in *Proceedings of the 10th International Conference on Latent Variable Analysis and Signal Separation*, 2012, pp. 34–41. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-28551-6˙5

[27] C. Joder, F. Weninger, F. Eyben, D. Virette, and B. Schuller, "Real-time speech separation by semi-supervised nonnegative matrix factorization," in *Proceedings of the 10th International Conference on Latent Variable Analysis and Signal Separation*, 2012, pp. 322–329. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-28551-6˙40

[28] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1118–1133, 2012.

[29] M. Spiertz and V. Gnann, "Note clustering based on 2D source-filter modeling for underdetermined blind source separation," in *Proceedings of the AES 42nd International Conference on Semantic Audio*, 2011, pp. 1–8.

[30] S. Araki, F. Nesta, E. Vincent, Z. Koldovsky, G. Nolte, A. Ziehe, and A. Benichoux, "The 2011 signal separation evaluation campaign (SiSEC2011): - audio source separation -," in *Proceedings of the 10th International Conference on Latent Variable Analysis and Signal Separation*, 2012, pp. 414–422. [Online]. Available: http://hal.inria.fr/hal-00655394

[31] P. L. Bartlett, E. Hazan, and A. Rakhlin, "Adaptive online gradient descent," in *Advances in Neural Information Processing Systems 20*, 2008, pp. 65–72. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-82.pdf

[32] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of the 19th International Conference on Computational Statistics*, 2010, pp. 177–186. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-7908-2604-3˙16

[33] L. Chen and J. Wang, "Dictionary learning with weighted stochastic gradient descent," in *Proceedings of the 2012 International Conference on Computational Problem-Solving*, 2012, pp. 9–12.

[34] L. S. R. Simon and E. Vincent, "A general framework for online audio source separation," in *Proceedings of the 10th International Conference on Latent Variable Analysis and Signal Separation*, 2012, pp. 397–404.

[35] Y. Salaun, E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry, D. Tran, and F. Bimbot, "The flexible audio source separation toolbox version 2.0," May 2014. [Online]. Available: http://hal.inria.fr/hal-00957412

[36] Y.-H. Yang, "On sparse and low-rank matrix decomposition for singing voice separation," in *Proceedings of the 20th ACM International Conference on Multimedia*, ser. MM '12. New York, NY, USA: ACM, 2012, pp. 757–760. [Online]. Available: http://doi.acm.org/10.1145/2393347.2396305

[37] L. Le Magoarou, A. Ozerov, and N. Q. K. Duong, "Text-informed audio source separation using nonnegative matrix partial co-factorization," Sep. 2013. [Online]. Available: http://hal.inria.fr/hal-00870066

[38] N. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1830–1840, 2010.

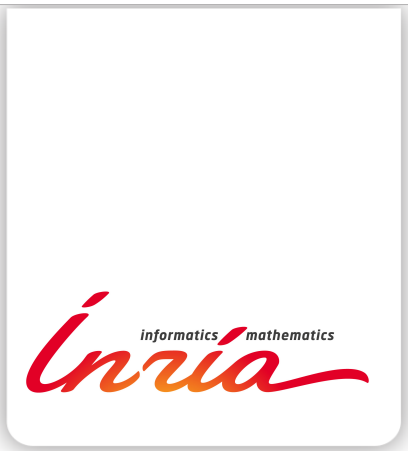
[39] M. Togami, "Online speech source separation based on maximum likelihood of local gaussian modeling," in *Proc. 2011 IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 2011, pp. 213–216.

[40] E. Vincent, S. Araki, F. Theis, G. Nolte, P. Bofill, H. Sawada, A. Ozerov, V. Gowreesunker, D. Lutter, and N. Q. Duong, "The signal separation evaluation campaign (2007–2010): Achievements and remaining challenges," *Signal Processing*, vol. 92, no. 8, pp. 1928–1936, Aug. 2012. [Online]. Available: http://hal.inria.fr/inria-00630985/en

[41] M. Desnoues, J. L. Durrieu, F. Thomas, G. Richard, O. Le Blouch, and E. Vincent, "QUASI database — a musical audio signal database for source separation." [Online]. Available: http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/

[42] E. Vincent, H. Sawada, P. Bofill, S. Makino, and J. P. Rosca, "First stereo audio source separation evaluation campaign: data, algorithms and results," in *Proc. 2007 Int. Conf. on Independent Component Analysis and Blind Source Separation*, 2007, pp. 552–559.

[43] C. Fevotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, Jun. 2011. [Online]. Available: http://dx.doi.org/10.1162/NECO˙a˙00168



RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399