

Reverberant Audio Source Separation via Sparse and Low-Rank Modeling

Simon Arberet, Pierre Vandergheynst

Abstract—

The performance of audio source separation from underdetermined convolutive mixture assuming known mixing filters can be significantly improved by using an analysis sparse prior optimized by a reweighting ℓ_1 scheme and a wideband data-fidelity term, as demonstrated by a recent article. In this letter, we show that the performance can be improved even more significantly by exploiting a low-rank prior on the source spectrograms. We present a new algorithm to estimate the sources based on i) an analysis sparse prior, ii) a reweighting scheme so as to increase the sparsity, iii) a wideband data-fidelity term in a constrained form, and iv) a low-rank constraint on the source spectrograms. Evaluation on reverberant music mixtures shows that the resulting algorithm improves state-of-the-art methods by more than 2 dB of signal-to-distortion ratio.

I. INTRODUCTION

An audio recording can be viewed as a mixture of several audio signals (e.g., musical instruments or speech), called *sources*. Mathematically, a *convolutive* mixture of N audio sources on M channels can be written as:

$$x_m(t) = \sum_{n=1}^N (a_{mn} \star s_n)(t) + e_m(t), \quad 1 \leq m \leq M, \quad (1)$$

where \star is the convolution operator, $s_n(t) \in \mathbb{R}$ and $x_m(t) \in \mathbb{R}$ denote sampled time signals of respectively the n -th source and the m -th mixture (t being a discrete time index), $a_{mn}(t) \in \mathbb{R}$ denotes the filter that models the impulse response between the n -th source and the m -th microphone, and $e_m(t)$ is the noise at the m -th microphone.

The goal of the *Blind Source Separation (BSS)* problem is to estimate the N source signals $s_n(t)$ ($1 \leq n \leq N$), given the M mixture signals $x_m(t)$ ($1 \leq m \leq M$). When the number of sources is larger than the number of mixture channels ($N > M$), the BSS problem is said to be *underdetermined* and is often addressed by sparsity-based approaches [1]–[3].

Audio signals are usually not sparse in the time domain, but they are in the time-frequency (TF) domain. Some approaches penalise the source TF coefficients with a ℓ_0 constraint (binary masking) [2], or a ℓ_1 cost [1], [4]. Another recent approach is the reweighting ℓ_1 scheme [5], which promotes a stronger sparsity assumption than the ℓ_1 cost, and has recently been shown to outperform ℓ_1 for source separation by almost 1 dB [6]. While *synthesis* sparse priors have been widely used for source modeling, *analysis* sparse priors have been used only

recently in audio source separation [6], and results showed that it improves the separation by about 1 dB in SDR.

Low-rank modeling, which can be traced back from Eckart [7] has been widely exploited in problems such as matrix completion [8] and robust PCA [9]. The idea of modeling the source spectrograms (i.e. the magnitude of the source TF coefficients) with a low-rank matrix has not been used directly, but indirectly via the non-negative matrix factorization (NMF) [10], [11] which also assumes the non-negativity of the factors. While this idea has been quite successful in audio BSS, it remains that the NMF approximation has some important limitations: its solution is non-unique and it converges but only to a fix point and very slowly. However, without these non-negativity constraints, the low-rank approximation, in the least squares sense, is unique and has a closed form solution, which can be computed via a singular value decomposition (SVD).

In this article, we focus on addressing the source estimation task, i.e. the second stage of a typical BSS approach, assuming that the mixing filters a_{mn} are known. The main contribution of this paper is to: i) introduce, in addition to a sparsity assumption, a low-rank model of the source spectrograms, i.e. we assume that the magnitude (and not the phase) of the short-time Fourier representation of each source is low-rank, and ii) derive an optimization algorithm based on a proximal splitting scheme [12] so as to estimate the sources. This algorithm also incorporates three ingredients, which were recently introduced in audio BSS [6]: i) an analysis sparsity prior, ii) a reweighting ℓ_1 scheme, and iii) a wideband data fitting constraint.

II. NOTATIONS

A. The convolutive mixture model in operator form

The mixture model (1) can be written as:

$$\mathbf{x} = \mathcal{A}(\mathbf{s}) + \mathbf{e}. \quad (2)$$

where $\mathbf{x} \in \mathbb{R}^{M \times T}$ is the matrix of the mixture composed of the $x_m(t)$ entries, i.e. $\mathbf{x} = [x_m(t)]_{m=1, t=1}^{M, T}$, T being the number of samples. Similarly $\mathbf{s} \in \mathbb{R}^{N \times T}$ is the matrix of sources composed of the $s_n(t)$ entries, $\mathbf{e} \in \mathbb{R}^{M \times T}$ is the matrix of the noise composed of the $e_m(t)$ entries, and $\mathcal{A} : \mathbb{R}^{N \times T} \rightarrow \mathbb{R}^{M \times T}$ is the discrete linear operator defined by

$$[\mathcal{A}(\mathbf{s})]_{m,t} = \sum_{n=1}^N (a_{mn} \star s_n)(t).$$

The adjoint operator $\mathcal{A}^* : \mathbb{R}^{M \times T} \rightarrow \mathbb{R}^{N \times T}$ of \mathcal{A} is obtained by applying the convolution mixing process with the adjoint filters $a_{nm}^*(t) \triangleq a_{mn}(-t)$, $\forall t$ instead of a_{mn} , that is: $[\mathcal{A}^*(\mathbf{x})]_{n,t} = \sum_{m=1}^M (a_{nm}^* \star x_m)(t)$.

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

The authors are with the Signal Processing Laboratory, Electrical Engineering Department, École Polytechnique Fédérale de Lausanne (EPFL), Station 11, CH-1015 Lausanne, Switzerland.
(e-mail: simon.arberet@gmail.com)

B. Time-frequency transform

As stated in the introduction, a powerful assumption is the sparsity of the audio sources in the TF domain. A popular TF representation is obtained via the short time Fourier transform (STFT).

The monochannel STFT operator $\psi : \mathbb{R}^T \rightarrow \mathbb{C}^{Q \times F}$ transforms a monochannel signal \mathbf{s}_n of length T , into a matrix $\psi(\mathbf{s}_n) = [\hat{\mathbf{s}}_n(qL/R, f)]_{q=1, f=1}^{Q, F} \in \mathbb{C}^{Q \times F}$ of TF coefficients $\hat{\mathbf{s}}_n(t, f)$, with $t = qL/R$, L being the window size, R the redundancy ratio, q and f , the time frame and frequency index, respectively. Let us also define the multichannel STFT operator $\Psi \in \mathbb{C}^{T \times B}$ that transforms a multichannel signal \mathbf{s} of length T , into a matrix $\tilde{\mathbf{s}} \in \mathbb{C}^{N \times B}$ populated by the $B = QF$ TF column vectors $\hat{\mathbf{s}}(t, f) \in \mathbb{C}^N$. Thus $\tilde{\mathbf{s}} = \mathbf{s} \Psi$, and the ISTFT is obtained by applying the adjoint operator $\Psi^* \in \mathbb{C}^{B \times T}$ on the STFT coefficients $\tilde{\mathbf{s}}$, i.e. $\mathbf{s} = \tilde{\mathbf{s}} \Psi^*$. With these notations, it is clear that $\mathbf{s}_n \Psi = \text{vec}(\psi(\mathbf{s}_n))$, where $\text{vec}(\cdot)$ is the vec operator which maps a matrix into a vector by stacking its columns. Let also define the *source spectrogram* of source \mathbf{s}_n as $|\psi(\mathbf{s}_n)| \in \mathbb{R}_+^{Q \times F}$, where $|\cdot|$ is the element wise absolute value.

III. PROBLEM FORMULATION

In order to estimate the sources from the mixture, we formulate an optimization problem composed of three terms. First, as we want our convolutive mixture model (2) to match the observations, we impose the reconstruction error $\|\mathbf{x} - \mathcal{A}(\mathbf{s})\|_2$ to be small and bounded by ϵ . Secondly, we assume an analysis sparse prior of the source TF representation, and thus we would like to minimize the ℓ_0 norm $\|\mathbf{s} \Psi\|_0$. Finally we assume that the rank of each source spectrogram $|\psi(\mathbf{s}_n)|$ is bounded by a small integer l .

This problem is NP because of the ℓ_0 norm and thus cumbersome for a problem of our size. However, the ℓ_0 norm can be replaced by a ℓ_1 norm, or for a sparser solution, by a sequence of weighted ℓ_1 minimizations $\|\mathbf{s} \Psi\|_{W,1}$ where $W \in \mathbb{R}_+^{N \times B}$ is a matrix with positive entries w_{ij} , and $\|\mathbf{z}\|_{W,1} \triangleq \sum_{i,j} w_{ij} |z_{ij}|$ is the weighted ℓ_1 norm [5]. Finally, the problem we want to solve, replacing the ℓ_0 norm with the weighting ℓ_1 norm is:

$$\begin{aligned} & \underset{\mathbf{s} \in \mathbb{R}^{N \times T}}{\text{argmin}} \|\mathbf{s} \Psi\|_{W,1} \\ & \text{subject to } \|\mathbf{x} - \mathcal{A}(\mathbf{s})\|_2 \leq \epsilon, \\ & \text{rank}(|\psi(\mathbf{s}_n)|) \leq l, \quad n = 1, \dots, N. \end{aligned} \quad (3)$$

This problem is still non-convex and hard to solve because the last constraint is non-convex¹. We will see later in the paper how to deal with it.

¹It is classical in convex optimization to relax the rank function by the nuclear norm in order to make the problem convex. However replacing the rank function with the nuclear norm in the last line of (3) will not make the problem convex because of the composition with the absolute value. Moreover, it is also more convenient to explicitly set the desired rank than having to tune a regularization parameter or to set a bound on the nuclear norm.

IV. OPTIMIZATION ALGORITHMS

In order to estimate the sources, an optimization algorithm called SSLR is derived. This (meta-)algorithm solves a sequence of optimization subproblems, each of which involves finding the solution of problem (3).

A. The SSRA and SSLR algorithms

The SSRA algorithm [6] is an iterative procedure which consists in computing, at each iteration k , the solution $\mathbf{s}^{(k)}$ of a weighted ℓ_1 problem, for a given weight matrix $W^{(k)}$, and then re-estimating W such that the weights $W^{(k+1)}$ are essentially the inverse of the value of the solution $\mathbf{s}^{(k)}$ of the current problem. This reweighing scheme is a classical procedure [5], [6], [13] which has been proved to approach the ℓ_0 norm minimization. In this paper we are using the same reweighing approach as SSRA, but with subproblem (3) instead of the weighted ℓ_1 problem of [6] which is essentially the same as problem (3) but without the low-rank constraints. We call SSLR the resulting procedure.

B. Convex optimization algorithms

At each iteration of the reweighing approach described in section IV-A, the solution of problem (3) has to be computed. In order to compute the solution of this problem, we rely on the framework of proximal splitting methods [12], because first they are efficient convex optimization algorithms that can deal with multiple (eventually non-smooth) terms and constraints, and secondly because they are particularly well suited for large scale problems and relatively easy to implement. While in Problem (3), the ℓ_2 -ball is a convex set, the set of low-rank matrices is non-convex. However, despite any convergence guaranty in general, using non-convex set constraints in proximal splitting methods can lead to efficient algorithms in practice when the projection can be computed exactly [14].

We first introduce the general framework of *proximal splitting methods*. Then we describe the PSDMM algorithm (Algorithm 2) which is a well-adapted algorithm to solve optimization problems involving an arbitrary number of non-smooth functions, and more particularly problem (3).

1) *Proximal splitting methods*: As we will see in section IV-B3, proximal splitting methods can solve optimization problems of the form:

$$\underset{\mathbf{s} \in \mathbb{R}^{N \times T}}{\text{argmin}} \sum_{i=1}^I f_i(\mathcal{L}_i(\mathbf{s})), \quad (4)$$

where f_i , are convex functions from \mathbb{R}^{J_i} to \mathbb{R} and $\mathcal{L}_i : \mathbb{R}^{N \times T} \rightarrow \mathbb{R}^{J_i}$ are bounded linear operators. Note that any convex constraint C on \mathbf{s} can be incorporated in this formulation via the indicator function $i_C(\cdot)$, where C represents the constraint set, and $i_C(\mathbf{s}) = 0$ if $\mathbf{s} \in C$, and $+\infty$ otherwise.

Problem (3) can be seen as a particular instance of problem (4) with three functions f_1, f_2, f_3 , and with $\mathcal{L}_1 = \mathcal{L}_3 = \mathbf{I}$, $\mathcal{L}_2 = \mathcal{A}$, $f_1(\mathbf{s}) = \|\mathbf{s} \Psi\|_{W,1}$, $f_2(\mathcal{A}(\mathbf{s})) = i_{\mathcal{B}_{\ell_2}^\epsilon}(\mathcal{A}(\mathbf{s}))$, where $\mathcal{B}_{\ell_2}^\epsilon = \{\mathbf{s} \in \mathbb{R}^{N \times T} : \|\mathbf{s} - \mathbf{x}\|_2 \leq \epsilon\}$, and $f_3(\mathbf{s}) = i_{\mathcal{R}^l}(\mathbf{s})$, where $\mathcal{R}^l = \{\mathbf{s} \in \mathbb{R}^{N \times T} : 1 \leq n \leq N, \text{rank}(|\psi(\mathbf{s}_n)|) \leq l\}$.

Algorithm 1: ADMM algorithm

Initialize: $k = 0$, $\mathbf{y}^{(0)} \in \mathcal{G}$, $\mathbf{z}^{(0)} \in \mathcal{G}$, $\gamma > 0$.
repeat
 $\mathbf{s}^{(k+1)} = \text{prox}_{\gamma \mathbf{L}}^{\mathbf{L}}(\mathbf{y}^{(k)} - \mathbf{z}^{(k)})$
 $\mathbf{y}^{(k+1)} = \text{prox}_{\gamma F}(\mathbf{L}(\mathbf{s}^{(k+1)}) + \mathbf{z}^{(k)})$
 $\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} + \mathbf{L}(\mathbf{s}^{(k+1)}) - \mathbf{y}^{(k+1)}$
 $k = k + 1$.
until convergence;
return $\mathbf{s}^{(k)}$

Note that $f_1(\mathbf{s})$ and $f_2(\mathcal{A}(\mathbf{s}))$ are convex, but $f_3(\mathbf{s})$ is not convex because \mathcal{R}^l is a non-convex set.

The key concept in proximal splitting methods is the use of the proximity operator prox_{f_i} of a function f_i defined as:

$$\text{prox}_{f_i}(\mathbf{z}) \triangleq \underset{\mathbf{y} \in \mathbb{R}^{J_i}}{\text{argmin}} f_i(\mathbf{y}) + \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2, \quad (5)$$

which is a natural extension of the notion of a projection. This definition extends naturally for some matrices \mathbf{z} and \mathbf{y} , by replacing the ℓ_2 norm with the Frobenius norm. Solution to (4) is reached iteratively by successive application of the proximity operator associated with each function f_i . See [12] for a review of proximal splitting methods and their applications in signal and image processing.

We derive in the appendix the proximity operators of functions $f_1(\mathbf{s}) = \|\mathbf{s} \Psi\|_{W,1}$, $f_2(\mathbf{s}) = i_{\mathcal{B}_{\ell_2}^\epsilon}(\mathbf{s})$ involved in optimization problem (3), and the projection on \mathcal{R}^l for function $f_3(\mathbf{s}) = i_{\mathcal{R}^l}(\mathbf{s})$ which can not formally be called "prox" because f_3 is not a convex function. We derive in the following sub-sections the optimization framework to solve problem (4).

2) *ADMM Algorithm*: The Alternating Direction Method of Multipliers (ADMM) [12] is a well suited algorithm to solve large-scale convex optimization of the form:

$$\underset{\mathbf{s} \in \mathcal{H}}{\text{argmin}} F(\mathbf{L}(\mathbf{s})) + G(\mathbf{s}), \quad (6)$$

where $F : \mathcal{G} \rightarrow]-\infty, +\infty]$ and $G : \mathcal{H} \rightarrow]-\infty, +\infty]$ are proper, convex, lowersemicontinuous (l.s.c.) functions, \mathcal{H} and \mathcal{G} being finite-dimensional real vector spaces equipped with an inner product $\langle \cdot, \cdot \rangle$, and a norm $\|\cdot\| = \langle \cdot, \cdot \rangle^{\frac{1}{2}}$. The map $\mathbf{L} : \mathcal{H} \rightarrow \mathcal{G}$ is a continuous linear operator with induced norm: $\|\mathbf{L}\| = \max\{\|\mathbf{L}(\mathbf{s})\| : \mathbf{s} \in \mathcal{H} \text{ with } \|\mathbf{L}(\mathbf{s})\| \leq 1\}$. If \mathbf{L} is injective, the ADMM algorithm described in Algorithm 1 converges to a solution of (6), where $\text{prox}_{\mathbf{L}}^{\mathbf{L}}$ is defined by:

$$\text{prox}_{\mathbf{L}}^{\mathbf{L}}(\mathbf{y}) \triangleq \underset{\mathbf{s} \in \mathcal{H}}{\text{argmin}} G(\mathbf{s}) + \frac{1}{2} \|\mathbf{L}(\mathbf{s}) - \mathbf{y}\|_2^2. \quad (7)$$

Minimization $\mathbf{s}^{(k+1)} = \text{prox}_{\mathbf{L}}^{\mathbf{L}}(\mathbf{y}^{(k)} - \mathbf{z}^{(k)})$ is a least squares problem including the linear operator \mathbf{L} which computation necessitates inner iterations. Antonin Chambolle and Thomas Pock [15] proposed a trick to precondition this step. Using their preconditioner (see section B in the Appendix), this minimization can be replaced by a simple prox computation, yielding the preconditioned ADMM algorithm also known as Chambolle-Pock Algorithm. Interestingly, the convergence

Algorithm 2: PSDMM: Preconditioned SDMM algorithm

Initialize: $k = 0$, $\mathbf{s}^{(0)} \in \mathbb{R}^{N \times T}$, for $i = 1, \dots, I$,
 $\mathbf{z}^{(i,0)} \in \mathbb{R}^{J_i}$, $\gamma > 0$, $\tau < \gamma / \|\mathbf{L}\|^2$
repeat
for $i \leftarrow 1$ **to** I **do**
 $\mathbf{y}^{(i,k+1)} = \text{prox}_{\gamma f_i}(\mathcal{L}_i(\mathbf{s}^{(k)}) + \mathbf{z}^{(i,k)})$
 $\mathbf{z}^{(i,k+1)} = \mathbf{z}^{(i,k)} + \mathcal{L}_i(\mathbf{s}^{(k)}) - \mathbf{y}^{(i,k+1)}$
 $\mathbf{s}^{(k+1)} = \mathbf{s}^{(k)} - \frac{\tau}{\gamma I} \sum_{i=1}^I \mathcal{L}_i^*(2\mathbf{z}^{(i,k+1)} - \mathbf{z}^{(i,k)})$
 $k = k + 1$.
until convergence;
return $\mathbf{s}^{(k)}$

of this algorithm has been proved [15] for a general (not necessarily injective) bounded linear operator \mathbf{L} .

3) *Preconditioned SDMM (PSDNN) Algorithm*: In a similar way as in [12], problem (4) can be formulated as a particular case of problem (6) in the I -fold product space $\mathcal{H} = \mathbb{R}^{N \times T} \times \dots \times \mathbb{R}^{N \times T}$, with $\mathcal{G} = \mathbb{R}^{J_1} \times \dots \times \mathbb{R}^{J_I}$. We denote $\mathbf{s} = (\mathbf{s}_1, \dots, \mathbf{s}_I)$ a generic element of \mathcal{H} , and $\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_I)$ a generic element of \mathcal{G} . Then we define $\mathbf{L} : \mathcal{H} \rightarrow \mathcal{G}$ by $\mathbf{L}(\mathbf{s}) = (\mathcal{L}_1(\mathbf{s}_1), \dots, \mathcal{L}_I(\mathbf{s}_I))$, $F(\mathbf{z}) = \sum_{i=1}^I f_i(\mathbf{z}_i)$, and $G(\mathbf{s}) = i_D(\mathbf{s})$ where, $i_D(\cdot)$ the indicator function of the convex set $D = \{(\mathbf{s}, \dots, \mathbf{s}) \in \mathcal{H} : \mathbf{s} \in \mathbb{R}^{N \times T}\}$. By deriving algorithm 1 with this parametrization and the Chambolle-Pock preconditioner, we obtain algorithm 2, denoted PSDNN.

V. EXPERIMENTS

We evaluated our SSLR algorithm with state-of-the-art methods over convolutive mixtures of music sources. For all the experiments, the test signals are sampled at 11 kHz and we use a STFT with cosine windows.

A. Experimental protocol

The mixing filters were room impulse responses simulated via the Roomsim toolbox [16], with a room size of dimension $3.55 \text{ m} \times 4.45 \text{ m} \times 2.5 \text{ m}$, and with the same microphones and source configuration as in [4]. The number of microphones was $M = 2$, and the number of sources was varied in the range $3 \leq N \leq 6$. The distances of the sources from the center of the microphone pairs was varied between 80 cm and 1.2 m. The mixing filters were generated with a reverberation time RT_{60} of 250 ms, and a microphone spacing of one meter. For each case $N = 3$ to 6, ten mixtures were realized by convolving, for each mixture, M mixing filters with N music sources of the BSS Oracle dataset composed of 30 music signals. For all the constrained methods, we set $\epsilon = 10^{-4}$, and we vary the low-rank parameter from $l = 5$ to $l = 30$. We also compared our algorithm with the classical DUET method [2] as well as SSRA [6] and the synthesis- ℓ_1 minimization with wideband data-fidelity (BPDN-S) [4], [6]. We initialized all the methods that need initialization, by applying the adjoint mixing operator to the mixture signal, i.e. $\mathbf{s}^{(0)} := \mathcal{A}^*(\mathbf{x})$.

The performance is evaluated for each source using the signal-to-distortion ratio (SDR), as defined in [18], which indicates the overall quality of each estimated source compared

to the target. We then average this measure over all the sources and all the mixtures for each mixing condition.

B. Results

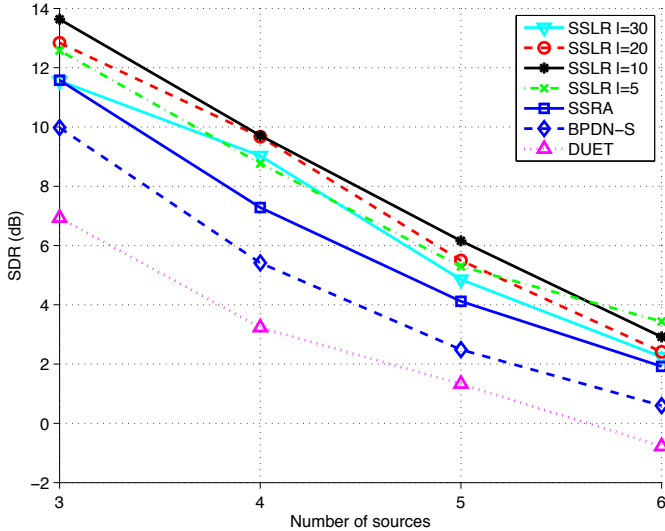


Fig. 1. Variation of the average SDR as a function of the number N of sources over music mixtures with reverberation time $RT_{60} = 250$.

The results are depicted in Fig. 1. We can notice that the best performance is achieved with our proposed SSLR method with a maximal rank of $l = 10$. The improvement with respect to SSRA is about 2 ± 1 dB in SDR depending of the number of sources. This shows the relevance of the low-rank constraint to model the source spectrograms. Moreover, all the other versions of SSLR, with other rank constraints l , outperformed SSRA, which indicates that the low-rank constraint does not degrade the performance even when l is not set optimally.

VI. CONCLUSION

We proposed a novel algorithm for reverberant audio source separation, which exploits the structure of the sources via a (analysis) sparse and low-rank prior on the source spectrograms. The sources are estimated via an optimization algorithm derived from the ADMM proximal scheme and the Chambolle-Pock preconditioner. The algorithm is also based on a reweighing analysis ℓ_1 approach so as to increase the sparsity and a wideband data-fidelity term in a constrained form. The results on convolutive music mixtures show that the proposed method outperforms all of the tested methods with an improvement of 2 ± 1 dB of SDR over SSRA, and 5 ± 1.5 dB over DUET. An extension of this work would be, in addition to the sources estimation, to estimate the mixing filters, possibly with an alternating optimization approach. Also it would be interesting to explore other variants of the problem and the algorithm.

APPENDIX

A. Proximity operators

We derive the proximity and projection operators for the functions $f_1(\mathbf{s}) = \|\mathbf{s}\Psi\|_{W,1}$ and $f_2(\mathbf{s}) = i_{\mathcal{B}_{\ell_2}^\epsilon}(\mathbf{s})$, and $f_3(\mathbf{s}) = i_{\mathcal{R}^l}(\mathbf{s})$ introduced in section IV-B.

Proposition 1. (Prox of $f_1(\cdot) = \|\cdot\Psi\|_{W,1}$) Let $\tilde{\mathbf{z}} \in \mathbb{C}^{N \times B}$ and $\mathbf{z} \in \mathbb{R}^{N \times T}$. If $\Psi \in \mathbb{C}^{T \times B}$ is a tight frame, i.e. $\Psi\Psi^* = \nu\mathbf{I}$, and $W \in \mathbb{R}_+^{N \times B}$ is a matrix of positive weights w_{ij} , then

$$\text{prox}_{\|\cdot\Psi\|_{W,1}}(\mathbf{z}) = \mathbf{z} + \nu^{-1}(\text{prox}_{\nu\|\cdot\|_{W,1}} - \mathbf{I})(\mathbf{z}\Psi)\Psi^*, \quad (8)$$

with

$$\text{prox}_{\nu\|\cdot\|_{W,1}}(\tilde{\mathbf{z}}) = (\text{prox}_{\nu w_{ij}|\cdot|}(\tilde{z}_{ij}))_{1 \leq i \leq N, 1 \leq j \leq B}, \quad (9)$$

where $\text{prox}_{\nu w_{ij}|\cdot|}$ is the soft thresholding operator given by $\text{prox}_{\lambda|\cdot|}(z_i) = \frac{z_i}{|z_i|}(|z_i| - \lambda)^+$ with $\lambda = \nu w_{ij}$ and $(\cdot)^+ = \max(0, \cdot)$.

The proof of this proposition can be found in [6].

Proposition 2. (Prox of $f_2(\cdot) = i_{\mathcal{B}_{\ell_2}^\epsilon}(\cdot)$, i.e. $\mathcal{P}_{\mathcal{B}_{\ell_2}^\epsilon}(\cdot)$)

$$\mathcal{P}_{\mathcal{B}_{\ell_2}^\epsilon}(\mathbf{z}) = \mathbf{x} + \min(1, \epsilon/\|\mathbf{z} - \mathbf{x}\|_2)(\mathbf{z} - \mathbf{x}). \quad (10)$$

Proposition 3. (Projection $\mathcal{P}_{\mathcal{R}^l}(\cdot)$ for $f_3(\cdot) = i_{\mathcal{R}^l}(\cdot)$)

$$\mathcal{P}_{\mathcal{R}^l}(\mathbf{z}) = \left(\mathcal{P}_{\mathcal{C}_l}(|\psi(\mathbf{z}_n)|) \circ e^{i\angle\psi(\mathbf{z}_n)} \right)_{1 \leq n \leq N}, \quad (11)$$

with $e^{i\angle\mathbf{z}} : \mathbf{z} \mapsto \mathbf{y} = e^{i\angle\mathbf{z}}\mathbf{z}$ being the element wise phase such that $y_{nm} = e^{i\arg(z_{nm})}$, and $\mathcal{P}_{\mathcal{C}_l}(\mathbf{z})$ being the projection onto the (non-convex) set $\mathcal{C}_l = \{\mathbf{z} : \text{rank}(\mathbf{z}) \leq l\}$ of matrices having a rank less or equal than l , which closed form solution, given by the Eckart-Young theorem [7] is: $\mathcal{P}_{\mathcal{C}_l}(\mathbf{z}) = \mathbf{u}\Sigma^l\mathbf{v}^*$, where $\mathbf{z} = \mathbf{u}\Sigma\mathbf{v}^*$ is the singular value decomposition (SVD) of \mathbf{z} and Σ is a diagonal matrix with non-increasing entries Σ_{ii} , and $\Sigma_{ii}^l := \begin{cases} \Sigma_{ii} & \text{if } i \leq l \\ 0 & \text{if } i > l. \end{cases}$

Proof: Let \mathcal{E}_l be the set of complex matrices which element-wise magnitude is a low-rank matrix, i.e. $\mathcal{E}_l = \{\mathbf{z} : \text{rank}(|\mathbf{z}|) \leq l\}$ and let $\mathcal{P}_{\mathcal{E}_l}(\mathbf{z}) = \text{argmin}_{\mathbf{y}} \{\|\mathbf{y} - \mathbf{z}\|_F : \mathbf{y} \in \mathcal{E}_l\}$ be the projection onto \mathcal{E}_l . For any matrices \mathbf{z} and \mathbf{y} , we have

$$\begin{aligned} \|\mathbf{y} - \mathbf{z}\|_F^2 &= \|\mathbf{y}\|_F^2 + \|\mathbf{z}\|_F^2 - 2\text{tr}\left(|\mathbf{z}|^\top \left(|\mathbf{y}| e^{i(\angle\mathbf{y} - \angle\mathbf{z})}\right)\right) \\ &\geq \|\mathbf{y}\|_F - \|\mathbf{z}\|_F^2. \end{aligned} \quad (12)$$

Inequality (12) is an equality when $\angle\mathbf{y} = \angle\mathbf{z}$. Thus, if the phase of \mathbf{y} is not constrained as in the set \mathcal{E}_l , the matrix \mathbf{y} minimizing $\|\mathbf{y} - \mathbf{z}\|_F$ is the one minimizing $\|\mathbf{y}\|_F - \|\mathbf{z}\|_F^2$ with $\angle\mathbf{y} = \angle\mathbf{z}$. Then, $\mathcal{P}_{\mathcal{E}_l}(\mathbf{z}) = \text{argmin}_{\mathbf{y}} \{\|\mathbf{y}\|_F - \|\mathbf{z}\|_F : \mathbf{y} \in \mathcal{C}_l, \angle\mathbf{y} = \angle\mathbf{z}\} = \mathcal{P}_{\mathcal{C}_l}(|\mathbf{z}|) \circ e^{i\angle\mathbf{z}}$.

B. Chambolle-Pock preconditioner [15]

The s -update step:

$$\begin{aligned} \mathbf{s}^{(k+1)} &= \text{prox}_{\gamma G}^{\mathbf{L}}(\mathbf{y}^{(k)} - \mathbf{z}^{(k)}) \\ &\triangleq \text{argmin}_{\mathbf{s} \in \mathcal{H}} \gamma G(\mathbf{s}) + \frac{1}{2} \|\mathbf{L}(\mathbf{s}) - (\mathbf{y}^{(k)} - \mathbf{z}^{(k)})\|^2 \end{aligned} \quad (13)$$

in the ADMM Algorithm 1 is a least squares problem including the linear operator \mathbf{L} which computation necessitates inner iterations. The Chambolle-Pock preconditioner consists in adding, in the minimization (13), the following term: $\frac{1}{2} \left\langle \left(\frac{1}{\tau} - \frac{1}{\gamma} \mathbf{L} \mathbf{L}^* \right) (\mathbf{s} - \mathbf{s}^{(k)}), \mathbf{s} - \mathbf{s}^{(k)} \right\rangle$, with $\tau < \frac{\gamma}{\|\mathbf{L}\|^2}$. As a result the s -update step becomes:

$$\mathbf{s}^{(k+1)} = \text{prox}_{\tau G}(\mathbf{s}^{(k)} - \tau \mathbf{L}^*(\bar{\mathbf{s}}^{(k)})),$$

with $\bar{\mathbf{s}}^{(k)} = \frac{1}{\gamma}(2\mathbf{z}^{(k)} - \mathbf{z}^{(k-1)})$.

REFERENCES

- [1] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal processing*, vol. 81, no. 11, pp. 2353–2362, 2001.
- [2] O. Yilmaz and S. T. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Trans. on Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.
- [3] P. O'Grady, B. Pearlmutter, and S. Rickard, "Survey of sparse and non-sparse methods in source separation," *International Journal of Imaging Systems and Technology*, vol. 15, no. 1, pp. 18–33, 2005.
- [4] M. Kowalski, E. Vincent, and R. Gribonval, "Beyond the narrowband approximation: Wideband convex methods for under-determined reverberant audio source separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1818–1829, 2010.
- [5] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.
- [6] S. Arberet, P. Vandergheynst, R. Carrillo, J. Thiran, and Y. Wiaux, "Sparse reverberant audio source separation via reweighted analysis," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 21, no. 7, pp. 1391 – 1402, July 2013.
- [7] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.
- [8] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.
- [9] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Advances in neural information processing systems*, 2009, pp. 2080–2088.
- [10] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 18, no. 3, pp. 550–563, Mar. 2010.
- [11] S. Arberet, A. Ozerov, N. Duong, E. Vincent, R. Gribonval, F. Bimbot, and P. Vandergheynst, "Nonnegative matrix factorization and spatial covariance model for under-determined reverberant audio source separation," in *Int. Conf. on Information Sciences Signal Processing and their Applications (ISSPA)*, May 2010, pp. 1–4.
- [12] P. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pp. 185–212, 2011.
- [13] R. Carrillo, J. McEwen, D. Van De Ville, J. Thiran, and Y. Wiaux, "Sparsity averaging for compressive imaging," *Signal Processing Letters, IEEE*, vol. 20, no. 6, pp. 591–594, 2013.
- [14] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [15] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.
- [16] D. Campbell, K. Palomaki, and G. Brown, "Roomsim, a MATLAB simulation of shoebox room acoustics for use in teaching and research," *Computing and Information Systems*, vol. 9, no. 3, pp. 48–51, 2005.
- [17] E. Vincent, R. Gribonval, and M. D. Plumbley, "Oracle estimators for the benchmarking of source separation algorithms," *Signal Processing*, vol. 87, no. 8, pp. 1933–1950, 2007.
- [18] R. Gribonval, L. Benaroya, E. Vincent, C. Févotte *et al.*, "Proposals for performance measurement in source separation," 2003.