Solving Time Domain Audio Inverse Problems using Nonnegative Tensor Factorization

Cagdas ¸ Bilen, Alexey Ozerov, Patrick Pérez

Solving Time Domain Audio Inverse Problems using Nonnegative Tensor Factorization

Çağdaş Bilen, Alexey Ozerov, and Patrick Pérez

***Abstract*—Nonnegative matrix and tensor factorizations (NMF and NTF) are important tools for modeling nonnegative data, which gained increasing popularity in various fields, a significant one of which is audio processing. However there are still many problems in audio processing, for which the NMF (or NTF) model has not been successfully utilized. In this work we propose a new algorithm based on NMF (and NTF) in the short-time Fourier domain for solving a large class of audio inverse problems with missing or corrupted time domain samples. The proposed approach overcomes the difficulty of employing a model in the frequency domain to recover time domain samples with the help of probabilistic modeling. Its performance is demonstrated for the following applications: Audio declipping and declicking (never solved with NMF/NTF modeling prior to this work); Joint audio declipping/declicking and source separation (never solved with NMF/NTF modeling or any other method prior to this work); Compressive sampling recovery and compressive sampling-based informed source separation (an extremely low complexity encoding scheme that is possible with the proposed approach and has never been proposed prior to this work).**

I. Introduction

Nonnegative matrix factorization (NMF) [1] and nonnegative tensor factorization (NTF) [2] decompositions have recently found great success in applications to audio modeling, notably for source separation [3]–[5], compression [6], [7], music transcription [8], [9] and audio inpainting [10]–[12]. It is now well-established in the audio signal processing community that spectrograms of natural audio signals exhibit a low-rank NMF (or NTF in case of multi-source signals) structure. They are indeed composed of relatively few characteristic spectral patterns modulated in time (*e.g.*, harmonic combs) that are well approximated by rank-1 nonnegative matrices/tensors. Within all these applications the power-spectrograms of single-channel or multichannel audio signals (usually powers of their short-time Fourier transforms (STFT)) are decomposed using NMF or NTF models.

However, these methods address quite poorly the situations when some chunks or samples of audio signals are missing in time domain, as for example in the situations of audio declipping or declicking, as described in a general audio inpainting paper [13]. Indeed, the NMF/NTF-based audio inpainting methods [10]–[12] assume that the audio data is missing directly in the corresponding time-frequency domain, usually the STFT domain. This is in fact the most convenient situation since the modeling itself is formulated in the STFT domain, and thus it becomes quite easy to take properly into account the missing values. In the case of audio with missing samples in the time domain, one can convert the missing information into an STFT domain formulation by simply assuming that all the STFT frames corresponding to missing time samples are missing in entirety. However, this will often lead to the loss of a huge amount of available information. In the case of a clipped audio for example, every STFT frame may be clipped, thus this naive solution would lead to considering the whole signal to be missing, even though there is perhaps only 20 % of the signal that is clipped in the time domain. Another problem of NMF/NTF-based audio inpainting methods [10]–[12] which consider fully-missing STFT coefficients is that NMF/NTF models are phase-invariant and thus they only allow estimating the magnitudes of the missing coefficients. As a result, the phase information, which is very important for audio perceptual quality, still needs to be reconstructed somehow. A popular approach by Griffin and Lim [14] is usually used for the phase reconstruction, but it performs quite poorly in many situations. As an alternative, a so-called high resolution NMF (HR-NMF) approach was proposed [15], [16]. This approach extends the NMF to model temporal dependencies between time-frequency bins, which yields better phase estimates. However, for the moment this approach is quite computationally expensive and it is limited to harmonic sounds. At the same time, when some samples are missing in the time domain and one manages to estimate properly the phase-invariant NMF model and the missing samples from these observations, the resulting phase estimates should be better than those obtained via Griffin and Lim's approach [14], since missing samples in time domain does not mean completely discarding the phase information in the STFT domain.

In this work, we propose a new approach allowing the estimation of lost time domain audio samples of audio sources and/or their mixture via applying a low-rank NMF/NTF model to latent power-spectrograms of the signals in the time frequency domain. The proposed method uses Itakura Saito (IS) divergence [4] for measuring how well the given NMF/NTF model parameters estimate the signal variances while using all the information available from all of the known time domain samples from the sources and/or the mixture. The model parameters are estimated using a generalized expectation-maximization (GEM) algorithm [17] and Wiener filtering [18] is used to recover the unknown signals. Unlike some other approaches that directly apply NMF/NTF model on the STFT coefficient magnitudes or powers, the proposed

Ç. Bilen is with Audio Analytic Ltd., UK, e-mail: contact@cagdasbilen.com. A. Ozerov is with Technicolor, France, e-mail: alexey.ozerov@technicolor.com. P. Pérez is with Valeo.ai, France, e-mail: patrick.perez@valeo.com

This work was partially supported by ANR JCJC program MAD (ANR-14-CE27-0002).

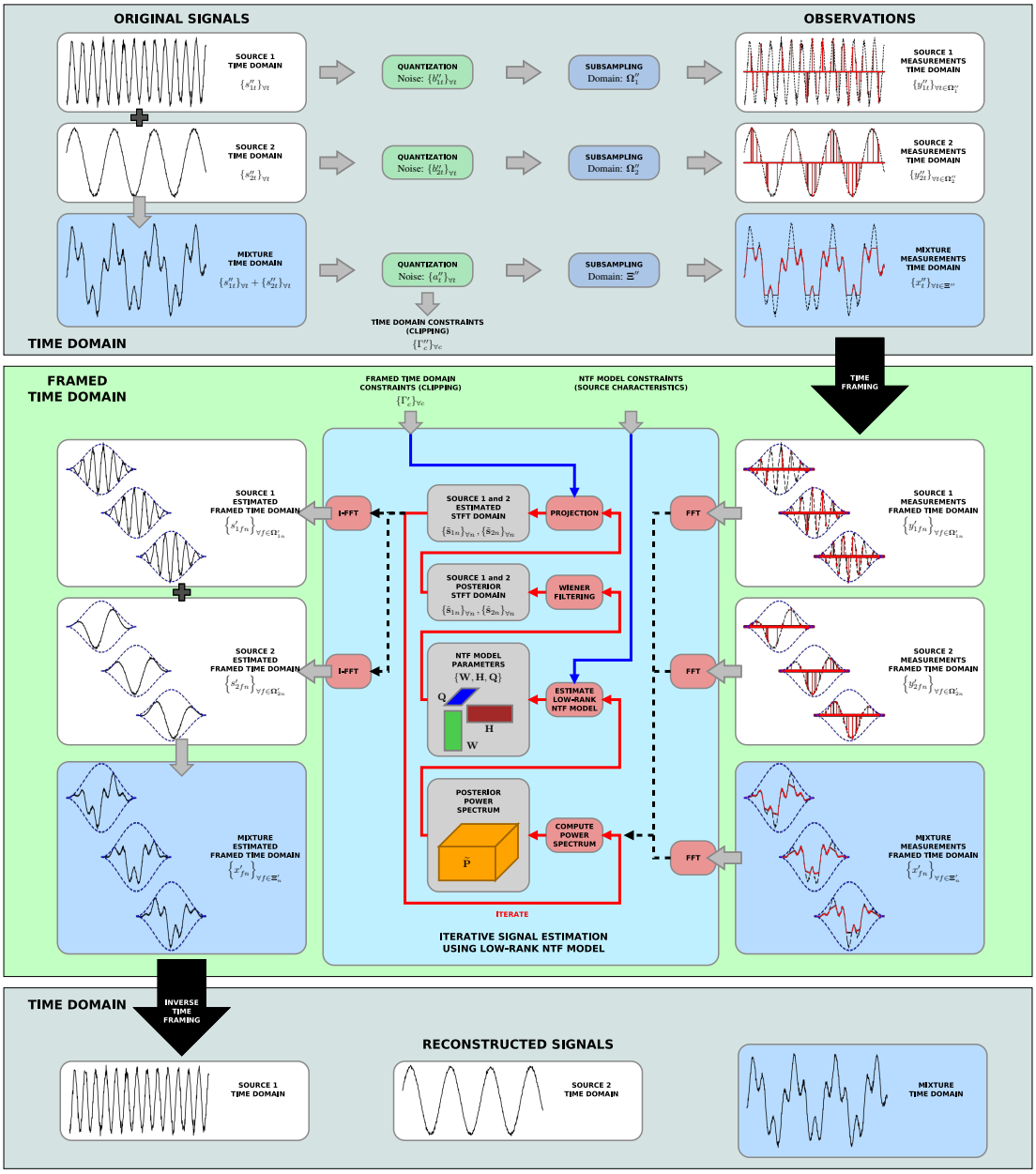


Fig. 1: The general framework of the proposed algorithm illustrating recovery of a mixture signal and its sources from a subset of the quantized samples of the sources and/or the mixture. The top section displays the time domain signals and an illustration of the generalized time domain audio inverse problem (recovering the sources from the measurements). The middle section illustrates the framed time domain variables, and the setup of the framed-time domain audio inverse problem (recovering the framed sources from the framed measurements). The middle section also illustrates a summary of the proposed algorithm steps. The bottom section illustrates the final output of the algorithm in time domain.

approach is formulated as a probabilistic Gaussian model on the complex-valued STFT coefficients. This enables us to estimate the NMF/NTF model in a *maximum likelihood (ML)* sense directly from the time domain observations, thus avoiding sub-optimally converting this missing information into the STFT domain. Furthermore, thanks to the flexibility of the NMF/NTF representations, the proposed framework can take into account mixtures of several sources, where both the sources and the mixtures can be partially or fully-missing in time domain. Last but not least, when the observed signals are not only partially lost but also corrupted, such as by noise or quantization, these corruptions can also be taken into account in the proposed approach. Within this general formulation the proposed framework is not limited to audio inpainting, but also becomes useful for different new applications related to audio compression, enhancement and source separation. This work builds on several previous conference/workshop publications [19]–[24] by the authors. Particular instances of the proposed approach for some specific applications have been presented in [19]–[21] and summarized in [22]. In this paper, we provide a

generalized formulation that is highly flexible to be adapted to various applications. We also present more comprehensive and extended experimental results, notably new experiments on compressive sampling recovery. The audio source separation approach presented in [23] falls within this general formulation as well, but it is not considered here for conciseness. Finally, while we here formulate the framework in the case of single-channel audio mixtures, its extension to the multichannel case is straightforward, which has been demonstrated in [24] for the declipping application.

More specifically, our general framework allowing recovering audio sources from partially observed and possibly quantized time domain audio samples of the sources and/or their mixture is applied here to the following existing or new applications:

- *Time domain audio inpainting and audio declipping* [13], [19], [25]–[28], where the mixture consisting of just one source is partially observed due to, *e.g.*, clipping. This is an existing application and we propose a new method to solve it.
- *Joint audio inpainting and source separation* [20], where the mixture consisting of several latent sources is partially observed due to, *e.g.*, clipping. The problem itself exists, but to the best of our knowledge, it was never addressed in a direct and systematic manner.
- *Compressive sampling recovery* [29], where the mixture consisting of just one source is partially observed due to a random sub-sampling. This is an existing application and we propose a new method to solve it.
- *Compressive sampling-based informed source separation* [21], where the mixture is observed and it consists of several latent sources that are partially observed after a random sub-sampling and quantization. This is a new informed source separation [7], [30] scheme resulting in an extremely fast encoder and a slow decoder.

The rest of this paper is organized as follows. The problem is formally defined in Section II and the proposed algorithm to solve it is described in Section III. Experiment results for various applications are given in Section IV and lastly final remarks and conclusions are presented in Section V. Readers willing to understand better and in detail the applications, before diving into the theoretical framework in Sections II and III, are invited to go through Section IV first.

II. Problem Definition

Let us consider a single-channel[1] mixture that is composed of J sources, among which each of the sources and/or the mixture might be fully, or partially observed and/or corrupted with noise (*e.g.*, quantization noise). For a mixture of length T, the mixture samples, x''_t,[2] are measured at a subset $\mathbf{\Xi}'' \subset [\![1, T]\!]$ of the entire time domain. Hence, the measured mixture samples can be represented in terms of unknown source samples, $s''_{jt}, j \in [\![1, J]\!]$, as

$$x''_t = \sum_{j=1}^{J} s''_{jt} + a''_t, \quad \forall t \in \mathbf{\Xi}'', \tag{1}$$

where a''_t represents the noise on the measurement sample due to various effects such as quantization. Furthermore, the individual sources may also be sampled at known subsets of the support $\mathbf{\Omega}''_j \subset [\![1, T]\!], j \in [\![1, J]\!]$ to obtain measured source samples, y''_{jt}, such that

$$y''_{jt} = s''_{jt} + b''_{jt}, \quad \forall t \in \mathbf{\Omega}''_j, \forall j \in [\![1, J]\!] \tag{2}$$

where b''_{jt} represents the noise for samples of each source. Lastly, for some problems such as declipping, we may also be given a set of C constraints, $\Gamma''_c(s''), c \in [\![1, C]\!]$, where each constraint, $\Gamma''_c(s'')$, is in one of the following forms:

$$s''_{j_c t_c} \geq \gamma''_c,\ s''_{j_c t_c} \leq \gamma''_c,\ \sum_{j=1}^{J} s''_{j t_c} \geq \gamma''_c,\ \sum_{j=1}^{J} s''_{j t_c} \leq \gamma''_c \tag{3}$$

in all of which γ''_c is a known constant and t_c and j_c are known time and source indices respectively.

Generalized Time Domain Audio Inverse Problem: *Given all of the above definitions, we define the generalized audio inverse problem in time domain as that of recovering the sources, $\{s''_{jt}\}_{\forall t,j}$ (and hence their mixture), given the noisy and incomplete measurements, $\{y''_{jt}\}_{\forall t \in \mathbf{\Omega}''_j, \forall j}$ and $\{x''_t\}_{\forall t \in \mathbf{\Xi}''}$, such that the constraints, $\{\Gamma''_c(s'')\}_{\forall c}$, are satisfied.*

III. Proposed Approach

A simple illustration of the known and unknown signals in the generalized time domain audio inverse problem is shown in the top section of the Figure 1, whereas the proposed algorithm in this work to solve this problem is illustrated in the middle section in the same figure. The individual steps of the proposed approach are explained in detail through the following subsections.

A. Redefining the Problem for a Frequency Domain Solution

The problem defined in Section II deals with constraints and unknowns in time domain, and as a result solving it with an approach that utilizes STFT domain constraints (such as the NTF model that will be introduced in Section III-B) can be computationally heavy and even intractable. To rectify this issue, we will introduce the framed-time domain and STFT domain notations, using which, we will define a modified problem that is much easier to handle.

The framed-time domain (or sometimes called windowed-time domain) is the representation of the time domain signal after it is split into (often overlapping) frames of fixed length, F, and multiplied by a fixed windowing function. Assuming that the total number of frames is N, the notations $x'_{fn}, y'_{jfn}, s'_{jfn}, a'_{fn}, b'_{jfn}, \mathbf{\Xi}'_n \subset [\![1, F]\!], \mathbf{\Omega}'_{jn} \subset [\![1, F]\!]$ represent the framed-time domain counterparts of the time domain notations defined in Section II for the source $j \in [\![1, J]\!]$, the intra-frame index $f \in [\![1, F]\!]$ within the frame $n \in [\![1, N]\!]$.

[1]For sake of simplicity, we only consider the single-channel case here. The proposed algorithm in this paper can be readily extended to multichannel case in a similar way as it is done in [24] for the declipping application.

[2]Throughout this paper the time domain signals will be denoted by letters with two primes, *e.g.*, x'', the framed-time domain signals by letters with one prime, *e.g.*, x', and complex-valued STFT coefficients by letters with no prime, *e.g.*, x.

The relationships between the framed-time domain variables are similar to that of the time domain counterparts such that[3]

$$x'_{fn} = \sum_{j=1}^{J} s'_{jfn} + a'_{fn}, \quad \forall f \in \Xi'_n, \forall n \tag{4}$$

$$y'_{jfn} = s'_{jfn} + b'_{jfn}, \quad \forall f \in \mathbf{\Omega}'_{jn}, \ \forall j, n \tag{5}$$

We represent the STFT coefficients of the source signals simply by $\{s_{jfn}\}_{\forall j,f,n}$. Note that, the STFT coefficients are simply the Fourier transforms of the framed-time domain signals, such that $\mathbf{s}_{jn} = [s_{jfn}]_{\forall f} = \mathbf{U}\mathbf{s}'_{jn}$ $\forall j, n$, where $\mathbf{s}'_{jn} \triangleq [s'_{jfn}]_{\forall f}$ and $\mathbf{U}$ is the normalized Fourier transform matrix satisfying $\mathbf{U}\mathbf{U}^H = \mathbf{U}^H\mathbf{U} = \mathbf{I}$.[4]

We now define a modified version of the initial problem using the framed-time domain variables and constraints, all of which can easily be computed from the time domain counterparts. This new definition of the problem has more relaxed conditions from the original problem in the sense that the problem is moved to a larger over-complete domain, and the correlation between the information within different frames is no longer defined. In the rest of this paper, we shall focus on solving this relaxed problem rather than the initial one.

Framed-Time Domain Audio Inverse Problem: *We define our problem as that of recovering the sources in framed-time domain, $\{s'_{jfn}\}_{\forall j,f,n}$ (or equivalently in STFT domain $\{s_{jfn}\}_{\forall j,f,n}$ since they are related with a unitary transform) given the noisy and incomplete framed-time measurements, $\{y'_{jfn}\}_{\forall f \in \mathbf{\Omega}'_{jn}, \forall j,n}$ and $\{x'_{fn}\}_{\forall f \in \Xi'_n, \forall n}$, such that the constraints, $\{\Gamma'_c(s')\}_{\forall c}$, are satisfied.*

B. Applying NTF Model estimated via a GEM Algorithm

In order to make the problem described in Section III-A easier to solve, we make a number of assumptions:

Assumption 1. ***The noise is independently Gaussian distributed with known variance:*** *The noise time samples for the observations, $\{a'_{jfn}\}_{\forall j,f,n}$ and $\{b'_{fn}\}_{\forall f,n}$, are independently distributed with zero mean Gaussian with known variances, $\{\sigma^2_{a,jfn}\}_{\forall j,f,n}$ and $\{\sigma^2_{b,fn}\}_{\forall f,n}$ respectively, i.e.*

$$a'_{jfn} \sim \mathcal{N}_c(0, \sigma^2_{a,jfn}), \quad b'_{fn} \sim \mathcal{N}_c(0, \sigma^2_{b,fn}), \quad \forall j, f, n. \tag{6}$$

Assumption 2. ***The sources are independently Gaussian distributed:*** *Similarly, the unknown STFT coefficients of the sources, $\{s_{jfn}\}_{\forall j,f,n}$, are also independently distributed with zero mean complex valued Gaussian with variance $\{v_{jfn}\}_{\forall j,f,n}$, i.e.*

$$s_{jfn} \sim \mathcal{N}_c(0, v_{jfn}), \quad \forall j, f, n. \tag{7}$$

Even though it is known that the noise in practice (such as quantization noise) is not always Gaussian, modeling the noise as Gaussian is still known to be a good enough approximation that provides significant computational advantage. Similarly the assumption of Gaussian distribution for the sources is also very common in audio community and accepted as a good approximation. It is noted when dealing with non-stationary signals that the assumption of gaussianity in the sources often results in very little loss in the source separation performance with the added benefit of much lower computational requirements [31]. Without further assumptions the variances v_{jfn} in (7) would be difficult to estimate, since there are as many parameters (variances) as the observations. Hence in this work we will also assume that the variances v_{jfn} are structured via a low-rank nonnegative tensor.

Assumption 3. ***Variances of the sources form a low rank NTF structure:*** *The tensor of source variances, $[v_{jfn}]_{j,f,n}$, is represented as the sum of few rank-1 nonnegative tensors, i.e.*

$$v_{jfn} = \sum_{k=1}^{K} q_{jk} w_{fk} h_{nk}, \quad \forall j, f, n \tag{8}$$

with number of components, K, sufficiently small. This so-called PARAFAC/CANDECOMP [32] NTF model can be parametrized by $\boldsymbol{\theta} = \{\mathbf{Q}, \mathbf{W}, \mathbf{H}\}$, such that $\mathbf{Q} = [q_{jk}]_{j,k} \in \mathbb{R}_+^{J\times K}$, $\mathbf{W} = [w_{fk}]_{f,k} \in \mathbb{R}_+^{F\times K}$ and $\mathbf{H} = [h_{nk}]_{n,k} \in \mathbb{R}_+^{N\times K}$.

The assumption of a low rank NTF structure on the joint variances of audio sources is well known in the audio source separation community and it is shown to be an accurate model for audio signals in practice [7], [30], [33]. Please note that when the signal is treated as a single source (*i.e.* without source separation and $J = 1$), the tensor of source variances reduces to a matrix and the decomposition is simply a low-rank NMF representation.

We can define the observed mixture vector at frame n, $\mathbf{x}'_n$, and the observed source vector at frame n for source j, $\mathbf{y}'_{jn}$, as

$$\mathbf{x}'_n \triangleq \left[x'_{fn}\right]_{\forall f \in \Xi'_n} \in \mathbb{R}^{|\Xi'_n|\times 1}, \tag{9}$$

$$\mathbf{y}'_{jn} \triangleq \left[y'_{jfn}\right]_{\forall f \in \mathbf{\Omega}'_{jn}} \in \mathbb{R}^{|\mathbf{\Omega}'_{jn}|\times 1}. \tag{10}$$

Hence for each frame we can define the observed data vector, $\mathbf{o}'_n$, and each unknown source vector, $\mathbf{s}'_{jn}$, as

$$\mathbf{o}'_n \triangleq \left[\mathbf{y}'^{\,T}_{1n}, \cdots, \mathbf{y}'^{\,T}_{Jn}, \mathbf{x}'^{\,T}_n\right]^T. \tag{11}$$

Given the three assumptions above, we propose estimating the NTF model $\boldsymbol{\theta}$ in the ML sense as

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}'} p(\{\mathbf{o}'_n\}_{\forall n} | \boldsymbol{\theta}'). \tag{12}$$

To achieve that we employ a GEM algorithm [17], while considering as latent data the totality of in general missing source STFT coefficients $\mathbf{S} = [s_{jfn}]_{\forall j,f,n}$. The algorithm iteratively alternates between an expectation step (E-step) for estimating the posterior power spectra of the signal and a maximization step (M-step) for updating the NTF model parameters. These two main steps can be summarized as follows:

- **E-step**: Estimate conditional expectations of source power spectra $|s_{jfn}|^2$, given the current model $\boldsymbol{\theta}$ and the

[3] From this point on, we shall use simply $\forall n$ to denote $\forall n \in [\![1, N]\!]$, $\forall f$ to denote $\forall f \in [\![1, F]\!]$ and $\forall j$ to denote $\forall j \in [\![1, J]\!]$, unless a subset of these sets is specified, *e.g.* Ξ'_n.

[4] $\mathbf{x}^T$ and $\mathbf{x}^H$ represent the non-conjugate transpose and the conjugate transpose of the vector (or matrix) $\mathbf{x}$ respectively.

observations:

$$\hat{p}_{jfn} = \mathbb{E}\left[\left|s_{jfn}\right|^2 \middle| \mathbf{o}'_n; \boldsymbol{\theta}\right], \quad \forall j, f, n. \tag{13}$$

- **M-step**: Re-estimate NTF model parameters such that the 3-valence tensor of the NTF model approximation, $\mathbf{V} = [v_{jfn}]_{\forall j,f,n}$, is as close to the 3-valence tensor of estimated source power spectra, $\widehat{\mathbf{P}} = [\hat{p}_{jfn}]_{\forall j,f,n}$, as possible with respect to the IS divergence [4]

$$D_{IS}(\widehat{\mathbf{P}}\|\mathbf{V}) = \sum_{\forall j,f,n} d_{IS}(\hat{p}_{jfn}\|v_{jfn}), \tag{14}$$

where $d_{IS}(x\|y) = x/y - \log(x/y) - 1$, and $\hat{p}_{jfn}$ and v_{jfn} are as specified respectively by (13) and (8).

The details of the E-step and the M-step are given in Sections III-C and III-D. In certain problems additional steps might also be required to satisfy certain constraints for the time domain signal or the NTF model parameters. It is described in Section III-E how these additional constraints can be handled by the proposed algorithm. A summary of the overall algorithm is given in Algorithm 1.

C. E-Step: Estimating Posterior Statistics

Following our assumptions of independently Gaussian distributed signals, we can write the posterior distribution of each source frame $\mathbf{s}_{jn}$ given the corresponding observed data $\mathbf{o}'_n$ and the NTF model $\boldsymbol{\theta}$ (or equivalently $\mathbf{V} = [v_{jfn}]_{\forall j,f,n}$ with v_{jfn} defined in (8)) as $\mathbf{s}_{jn}|\mathbf{o}'_n; \boldsymbol{\theta} \sim \mathcal{N}_c(\hat{\mathbf{s}}_{jn}, \widehat{\boldsymbol{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}})$ with $\hat{\mathbf{s}}_{jn}$ and $\widehat{\boldsymbol{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}$ being, respectively, posterior mean and posterior covariance matrix of the STFT coefficients, $\mathbf{s}_{jn}$. These terms can be computed respectively by Wiener filtering as [18]

$$\hat{\mathbf{s}}_{jn} = \boldsymbol{\Sigma}^H_{\mathbf{o}'_n\mathbf{s}_{jn}} \boldsymbol{\Sigma}^{-1}_{\mathbf{o}'_n\mathbf{o}'_n} \mathbf{o}'_n, \tag{15}$$

$$\widehat{\boldsymbol{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}} = \boldsymbol{\Sigma}_{\mathbf{s}_{jn}\mathbf{s}_{jn}} - \boldsymbol{\Sigma}^H_{\mathbf{o}'_n\mathbf{s}_{jn}} \boldsymbol{\Sigma}^{-1}_{\mathbf{o}'_n\mathbf{o}'_n} \boldsymbol{\Sigma}_{\mathbf{o}'_n\mathbf{s}_{jn}}, \tag{16}$$

given the definitions of the covariance matrices

$$\boldsymbol{\Sigma}_{\mathbf{o}'_n\mathbf{o}'_n} = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{y}'_{1n}\mathbf{y}'_{1n}} & \cdots & \underline{\mathbf{0}} & \boldsymbol{\Sigma}^H_{\mathbf{x}'_n\mathbf{y}'_{1n}} \\ \vdots & \ddots & \vdots & \vdots \\ \underline{\mathbf{0}} & \cdots & \boldsymbol{\Sigma}_{\mathbf{y}'_{Jn}\mathbf{y}'_{Jn}} & \boldsymbol{\Sigma}^H_{\mathbf{x}'_n\mathbf{y}'_{Jn}} \\ \boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{y}'_{1n}} & \cdots & \boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{y}'_{Jn}} & \boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{x}'_n} \end{bmatrix}, \tag{17}$$

$$\boldsymbol{\Sigma}_{\mathbf{o}'_n\mathbf{s}_{jn}} = \left[\underline{\mathbf{0}}^T_{L_{1,jn}\times F}, \boldsymbol{\Sigma}^T_{\mathbf{y}'_{jn}\mathbf{s}_{jn}}, \underline{\mathbf{0}}^T_{L_{2,jn}\times F}, \boldsymbol{\Sigma}^T_{\mathbf{x}'_n\mathbf{s}_{jn}}\right]^T, \tag{18}$$

$$\boldsymbol{\Sigma}_{\mathbf{y}'_{jn}\mathbf{y}'_{jn}} = \mathbf{U}(\boldsymbol{\Omega}'_{jn})^H \mathrm{diag}\left([v_{jfn}]_{\forall f}\right) \mathbf{U}(\boldsymbol{\Omega}'_{jn}) + \mathrm{diag}\left(\left[\sigma^2_{b,jfn}\right]_{\forall f\in\boldsymbol{\Omega}'_{jn}}\right), \tag{19}$$

$$\boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{x}'_n} = \mathbf{U}(\boldsymbol{\Xi}'_n)^H \mathrm{diag}\left(\left[\sum\nolimits_{\forall j} v_{jfn}\right]_{\forall f}\right) \mathbf{U}(\boldsymbol{\Xi}'_n) + \mathrm{diag}\left(\left[\sigma^2_{a,fn}\right]_{\forall f\in\boldsymbol{\Xi}'_n}\right), \tag{20}$$

$$\boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{y}'_{jn}} = \mathbf{U}(\boldsymbol{\Xi}'_n)^H \mathrm{diag}\left([v_{jfn}]_{\forall f}\right) \mathbf{U}(\boldsymbol{\Omega}'_{jn}), \tag{21}$$

$$\boldsymbol{\Sigma}_{\mathbf{y}'_{jn}\mathbf{s}_{jn}} = \mathbf{U}(\boldsymbol{\Omega}'_{jn})^H \mathrm{diag}\left([v_{jfn}]_{\forall f}\right), \tag{22}$$

$$\boldsymbol{\Sigma}_{\mathbf{x}'_n\mathbf{s}_{jn}} = \mathbf{U}(\boldsymbol{\Xi}'_n)^H \mathrm{diag}\left([v_{jfn}]_{\forall f}\right), \tag{23}$$

$$\boldsymbol{\Sigma}_{\mathbf{s}_{jn}\mathbf{s}_{jn}} = \mathrm{diag}\left([v_{jfn}]_{\forall f}\right), \tag{24}$$

where $\mathbf{U}(\boldsymbol{\Omega}'_{jn})$ is the $F \times |\boldsymbol{\Omega}'_{jn}|$ matrix of columns from $\mathbf{U}$ with index in $\boldsymbol{\Omega}'_{jn}$ and $L_{1,jn} \triangleq \sum_{l=1}^{j-1} |\boldsymbol{\Omega}'_{ln}|$, $L_{2,jn} \triangleq \sum_{l=j+1}^{J} |\boldsymbol{\Omega}'_{ln}|$. The term $\mathrm{diag}(\mathbf{x})$ represents a diagonal matrix with the vector $\mathbf{x}$ along the diagonal.

Finally, the posterior power spectra, $\widehat{\mathbf{P}} = [\hat{p}_{jfn}]_{\forall j,f,n}$ can be computed as

$$\hat{p}_{jfn} = \mathbb{E}\left[\left|s_{jfn}\right|^2 \middle| \mathbf{o}'_n; \boldsymbol{\theta}\right] = |\hat{s}_{jfn}|^2 + \widehat{\boldsymbol{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}(f,f). \tag{25}$$

D. M-step: Updating NTF Model Parameters

Estimating NTF model $\boldsymbol{\theta}$ in the ML sense is proven [4] equivalent to minimizing the IS divergence $D_{IS}(\widehat{\mathbf{P}}\|\mathbf{V})$ as defined in (14) between the tensor of variances, $\mathbf{V}$, and the given posterior power spectra tensor, $\widehat{\mathbf{P}}$.

A common optimization approach to estimate the model parameters, $\boldsymbol{\theta}$, that minimizes (14) is using multiplicative updates (MU) as described in [4]. In our case, starting from some initial nonnegative model parameters, the model parameters that minimize (14) can be found by applying several iterations of the following updates

$$q_{jk} \leftarrow q_{jk} \left(\frac{\sum_{f,n} w_{fk} h_{nk} \hat{p}_{jfn} v^{-2}_{jfn}}{\sum_{f,n} w_{fk} h_{nk} v^{-1}_{jfn}}\right), \tag{26}$$

$$w_{fk} \leftarrow w_{fk} \left(\frac{\sum_{n,j} q_{jk} h_{nk} \hat{p}_{jfn} v^{-2}_{jfn}}{\sum_{n,j} h_{nk} q_{jk} v^{-1}_{jfn}}\right), \tag{27}$$

$$h_{nk} \leftarrow h_{nk} \left(\frac{\sum_{f,j} q_{jk} w_{fk} \hat{p}_{jfn} v^{-2}_{jfn}}{\sum_{f,j} w_{fk} q_{jk} v^{-1}_{jfn}}\right). \tag{28}$$

In the beginning of the proposed GEM algorithm, the model parameters can be initialized randomly with nonnegative values. In the following iterations however, the update of the model parameters can be always applied starting from the current model parameters (instead of randomly initializing them before MU iterations each time $\widehat{\mathbf{P}}$ is updated).

E. Applying Additional Constraints

In many practical audio inverse problems, there may be additional knowledge on the signal to be estimated apart from the observed samples. We shall consider mainly two complementary types of knowledge on the signal to be treated, which provide:

i *Constraints on NTF model parameters*, such as some characteristic spectral patterns being active, some frequency or time bins being silent, or simply the frequency response being symmetric (time domain signal being real valued);

ii *Constraints on framed-time domain samples*, such as the constraints, $\{\Gamma'_c(s')\}_{\forall c}$, that were defined earlier.

The additional constraints on the model parameters, $\boldsymbol{\theta}$, are often easy to incorporate during the MU iterations or simply initializing them in a specific way. For instance, the symmetry in frequency (hence being real valued in time) can be enforced if the matrix $\mathbf{W}$ is updated to be always symmetrical along the frequency axis. Similarly if some of the characteristic spectral patterns are known *a priori* to be present in the sources, $\mathbf{W}$ can be initialized with a specific dictionary and then may never be updated to enforce using only these patterns. Another

example is, when certain entries of the matrices $\mathbf{W}, \mathbf{H}$ and $\mathbf{Q}$ are known to be zero, they can be simply initialized to be zero and these zero values will be automatically enforced in the following MU iterations. Lastly, in certain applications, it is even possible to change the model to enforce additional structures on the matrices $\mathbf{W}, \mathbf{H}$ and $\mathbf{Q}$, such as sparsity by small modifications on the MU equations [34].

Dealing with constraints on framed-time domain samples, unlike the constrains on the model parameters, is not straightforward. When a framed-time domain sample is known to be clipped or quantized, the original value of this sample is known to be above (below) a certain threshold or to lay within a certain interval, and the resulting posterior probability distribution of the sample is no longer Gaussian. As a result, estimating the posterior power spectrum with this modified probability distribution is not as simple as described in Section III-C. To overcome this problem, we propose to estimate the posterior power spectrum by computing the posterior mean, $\hat{\mathbf{s}}_{jn}$ and the posterior covariance, $\widehat{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}$, as described in Section III-C, but then projecting them so as to satisfy the time domain constraints to obtain modified statistics, $\tilde{\mathbf{s}}_{jn}$ and $\widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}$ respectively. As a result, the modified posterior power spectrum (to be used as the input for the NTF model update in M-step) is obtained as

$$\tilde{p}_{jfn} = |\tilde{s}_{jfn}|^2 + \widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}(f,f). \tag{29}$$

We define several approaches to compute the aforementioned modified statistics to satisfy the constraints in framed-time domain samples:

1) **Unconstrained:** The simplest way to perform the estimation is to ignore completely the constraints, treating the problem as a more generic audio inpainting in time domain. Hence during the iterations, the "constrained" signal is taken simply as the estimated signal, *i.e.* $\tilde{\mathbf{s}}_{jn} = \hat{\mathbf{s}}_{jn}, \forall n, j$, as is the posterior covariance matrix, $\widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}} = \widehat{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}, \forall n, j$.
2) **Ignored projection:** Another simple way to proceed is to ignore the constraint during the iterative estimation process and to enforce it at the end as a post-processing of the estimated signal. In this case, the signal is treated the same way as in the unconstrained case during the iterations.
3) **Signal projection:** A more advanced approach is to update the estimated signal at each iteration so that the magnitude obeys the constraints. As an example, let us suppose we have a constraint in the form $s'_{j_c f_c n_c} \geq \gamma'_c$ and it is not satisfied by the estimated posterior mean, *i.e.* $\hat{s}'_{j_c f_c n_c} < \gamma'_c$. We can simply set $\tilde{s}'_{j_c f_c n_c} = \gamma'_c$ and $\tilde{s}'_{jfn} = \hat{s}'_{jfn}$ for the rest of the support (and $\tilde{\mathbf{s}}_{jn} = \mathbf{U}\tilde{\mathbf{s}}'_{jn}$). Formally we can define,

$$\{\tilde{s}'_{jfn}\}_{\forall j,f,n} = \underset{\{z'_{jfn}\}_{\forall j,f,n}}{\operatorname{argmin}} \sum_{\forall j,f,n} |z'_{jfn} - \hat{s}'_{jfn}|^2 \quad \text{s.t.} \quad \{\Gamma'_c(z')\}_{\forall c} \tag{30}$$

Note that this approach does not update the posterior covariance matrix, *i.e.* $\widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}} = \widehat{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}, \forall n, j$.
4) **Covariance projection:** In order to update the posterior mean and the posterior covariance matrix in a consistent manner, we can re-compute the posterior mean and the posterior covariance by (15) and (16) respectively, by treating the projected signal samples in (30) at the support $\mathbf{\Omega}'_{\mathbf{m},jn} \triangleq \{f | \tilde{s}'_{jfn} \neq \hat{s}'_{jfn}\}$ as observed values for the current iteration. If the resulting estimation of the sources violates the time domain constraints on additional indices, those samples are also projected to obey the constraints and treated as observed. This process is repeated until a posterior mean, $\tilde{\mathbf{s}}_{jn}$, and a posterior covariance, $\widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}$, that are consistent with all the time domain constraints, are obtained. Note that in addition to updating the posterior covariance matrix, this approach also updates the entire posterior mean (or estimated signal) and not just the posterior mean at the indices of violated constraints.

Algorithm 1 GEM algorithm for solving Time Domain Audio Inverse Problems with NTF model

procedure RESTORE-AUDIO-WNTF
  Initialize nonnegative $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{H}, \mathbf{Q}\}$ randomly
  repeat
    E-step : Estimate $\hat{\mathbf{s}}_{jn}, \widehat{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}, \forall n, j$, given $\boldsymbol{\theta}$, $\mathbf{o}'_{jn}$ $\forall n, j$ ▷ see § III-C
    Time domain constraints : Estimate $\tilde{\mathbf{s}}_{jn}$, $\widetilde{\mathbf{\Sigma}}_{\mathbf{s}_{jn}\mathbf{s}_{jn}}$, $\forall n, j$ and $\widetilde{\mathbf{P}}$ given $\{\Gamma'_c\}_{\forall c}$ ▷ see § III-E
    M-step : Update $\boldsymbol{\theta}$ given $\widetilde{\mathbf{P}}$ ▷ see § III-D, § III-E
  until convergence criteria met
end procedure

IV. Important Applications and Experimental Results

The proposed algorithm is adapted to solve a number of audio inverse problems, some of which are explored for the first time in this work. For each of these problems, we performed a set of experiments on various audio examples and compared the performance to that of known state of the art algorithms when applicable.

In the experiments below, all the audio signals are sampled at 16 kHz, and the STFT within the various instances of the proposed algorithm is computed using a half-overlapping sine window of 1024 samples (64 ms).

A. Time Domain Audio Inpainting and Audio Declipping

The problem of recovering audio samples that are lost or corrupted is often called audio inpainting [13]. We use the term "time domain audio inpainting" to refer to the problems with missing or corrupted time domain audio samples as opposed to the audio inpainting problems with missing samples in the STFT domain, for which NMF/NTF models are already being used prominently [10]–[12]. We still prefer to differentiate these problems from "audio interpolation" since the missing samples might sometimes arrive in large gaps instead of being distributed over time, and sometimes we might even encounter time domain samples missing in conjunction with missing STFT coefficients as can be encountered with audio editing applications. Two specific instances of the time domain audio

inpainting problem are called audio declipping and audio declicking [13], in which one recovers the time domain audio samples that are lost due to clipping and clicking effects caused by audio recording and compression processes. The declipping problem in particular provides additional challenges with respect to the general time domain audio inpainting problem, because it often includes additional constraints for the time domain signal to be estimated. In the recent years, the models based on sparse, cosparse or group-sparse representations in certain dictionaries are shown to be performing best to solve these problems [13], [25]–[28]. Clipping and interpolation from a Bayesian perspective has been also addressed earlier [35]–[38], mostly relying on autoregressive modeling. Though recent approaches [13], [25]–[28] have been shown performing better than (or on par with) them (see, *e.g.*, [13], [25]). Despite the success of modeling audio signals with low rank NMF representations, the time domain audio inpainting problem, especially with the additional constraints as in audio declipping, is not trivial to solve with an NMF/NTF model in the time-frequency domain. This is possibly the main reason why these models have not been utilized in time domain audio inpainting problems successfully. The proposed approach can overcome this limitation and provides a new perspective on time domain audio signal recovery with equivalent or better performance than the state of the art.

In our experiments with the audio declipping problem, we consider an audio signal with no known source information (as such it is modeled as a single source, $J = 1$) that is clipped to a known threshold of magnitude $\tau > 0$. Thus the signal is accurately known for a subset of the support, $\bar{\Xi}''$, where signal magnitude is smaller than τ. For the remaining support, $\bar{\bar{\Xi}}'' = [\![1, T]\!] \setminus \bar{\Xi}''$, the signal is unknown but obeys the time domain constraints of the form,

$$\begin{array}{lll} s''_{t_c} \geq \tau, & \text{for} & x''_{\mathrm{c},t_c} > 0 \\ s''_{t_c} \leq -\tau, & \text{for} & x''_{\mathrm{c},t_c} < 0 \end{array}, \quad \forall t_c \in \bar{\bar{\Xi}}''. \tag{31}$$

where x''_{c,t_c} is the clipped signal. We also assume that there is no observation noise, *i.e.*, $\sigma^2_{a,jfn} = \sigma^2_{b,jfn} = 0$, $\forall j, f, n$, in (6).

In [28], various state of the art audio declipping algorithms are compared based on the experiments performed on music and speech examples. We have repeated these experiments using our approach with the same methodology and the datasets as reported in [28] and provided an overall comparison of our algorithm to the other approaches. The experiment procedure can be summarized as follows; 10 music and 10 speech signals, each of length of 4 seconds, are scaled to have maximum magnitude of 1 in time domain, and then artificially clipped at eight different clipping thresholds (uniformly spaced from 0.2 to 0.9). The proposed algorithm is tested with four different methods to handle the clipping constraints as described in Section III-E, namely *Unconstrained (NMF-U)*, *Ignored Projection (NMF-IP)*, *Signal Projection (NMF-SP)* and *Covariance Projection (NMF-CP)*. The music signals are declipped with 20 NMF components ($K = 20$), while 28 components are used for speech signals ($K = 28$). The proposed GEM algorithm is run for 50 iterations. The performance of the proposed algorithm is compared to five state of the art methods: iterative hard-thresholding (HT) [25], cosparsity (Cosp) [27], orthogonal matching pursuit (OMP) [13], social sparsity with empirical Wiener operator (SS-EW) and social sparsity with posterior empirical Wiener operator (SS-PEW) [28].

The performance metric that is used to compare the algorithms is the improvement of the signal to noise ratio (computed only on the clipped regions) with respect to the clipped signal, SNR_m, that is computed as [28]:

$$\mathrm{SNR}_m = 10 \log_{10} \frac{\sum_{\forall t \in \bar{\Xi}''} |x''_{\mathrm{o},t}|^2}{\sum_{\forall t \in \bar{\Xi}''} |x''_{\mathrm{o},t} - x''_{\mathrm{e},t}|^2}, \tag{32}$$

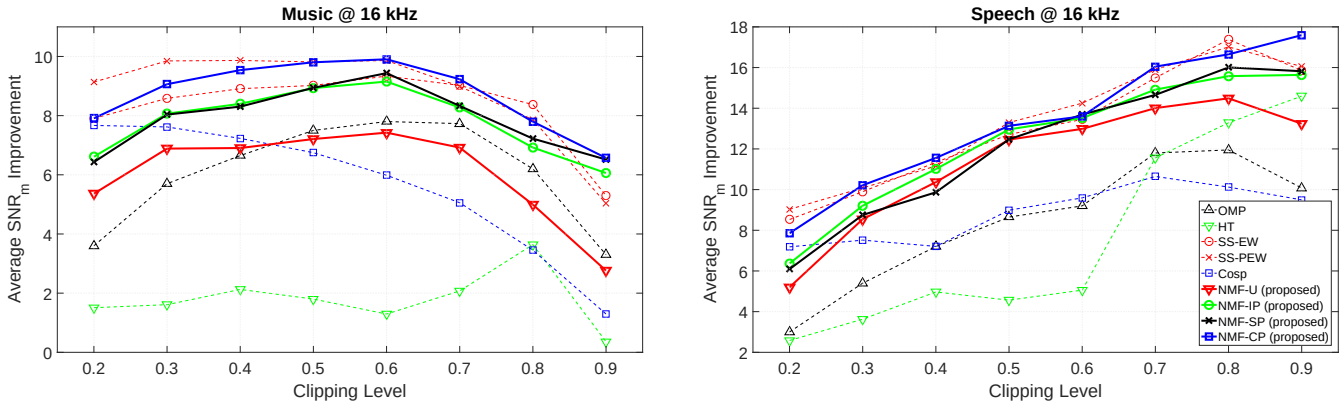
where $x''_{\mathrm{o},t}$ is the original time domain signal sample and $x''_{\mathrm{e},t}$ is the estimated signal sample. Finally, the performance is measured in terms of the SNR_m improvement, which is the difference between the SNR_m computed on the estimated signal and the SNR_m computed on the clipped signal.

The average performance of all the algorithms for declipping of music and speech signals is represented on Figure 2. It can be seen from the overall results that the proposed algorithm with the covariance projection (NMF-CP) has almost identical performance with the social sparsity based methods (SS-EW and SS-PEW) proposed in [28] while outperforming others. It can be also seen in the results that the model based algorithms (social sparsity and the NMF model) significantly outperform the methods relying on just sparsity (OMP and HT) or on just cosparsity (Cosp).

Regarding the effect of clipping constraints, the first thing to notice is that the performance of NMF-U with respect to NMF-IP (and NMF-SP) shows that simple constraints on the signal magnitude can noticeably improve the performance especially for music signals, hence they should not be ignored when possible. NMF-IP and NMF-SP are shown to have almost identical performance, even though the latter applies the constraints on the posterior mean of the signal at every iteration and the former simply applies a post processing to the final result. This observation combined with the superior performance of NMF-CP compared to the other methods demonstrates the importance of updating the posterior power spectrum more accurately for the success of the NMF-based methods.

Even though the performance is not better than the social sparsity approaches at first glance, the proposed algorithm has room for improvements in various aspects:

- NMF model can be easily extended to other, more structured NMF-like models such as source-excitation model or harmonic NMF [31]. As shown in [31] in case of source separation, having a specific model with structure that is well adapted to the considered class of signals (*e.g.*, speech, music, etc.) may improve the overall performance.
- It is shown in the results that the performance of our method depends significantly one the way the clipping constraint is handled. Therefore an alternative, more accurate computation of the posterior power spectrum might also improve the results further, whereas in dictionary based methods there is no approximation for the



(a) Average SNR$_m$ improvement computed over 10 music signals. (b) Average SNR$_m$ improvement computed over 10 speech signals.

Fig. 2: The average performance of all the audio declipping algorithms as a function of the clipping threshold (lower threshold corresponds to more severe clipping).

clipping constraints, hence performance improvement in this regard is not possible.

It should be noted that dealing with time domain constraints while enforcing a model on the STFT domain comes at a computational cost in the Wiener filtering stage of the proposed algorithm. Luckily, this step is independent for each frame of the signal and hence can be easily parallelized, *e.g.*, using graphical processing units (GPUs), to get a significant speed-up. On the other hand, estimating the signal independently within each window comes with the disadvantage that the estimation is not possible when there are no observed samples within a window. In practice, however, the loss of an entire window due to clipping is not probable for natural audio signals when the window size is chosen properly and the clipping threshold is not extremely low.

B. Joint Audio Inpainting and Source Separation

The audio source separation is a well known problem for which the NMF/NTF modeling in the time-frequency domain is shown to be quite successful [3]–[5]. However in all source separation problems, the audio mixture is assumed to be known perfectly whereas in practice the mixture can also have missing or corrupted (due to noise or quantization) samples in time domain. This joint problem naturally arises when one would like to perform source separation on a mixture that is degraded due to clipping effects or other degradations. This problem can also often arise in audio editing applications where some part of the audio is intentionally removed to suppress unwanted artefacts. Additionally one can also consider the case when source separation is not really needed, but a multi-source model is still employed to improve the performance of audio inpainting when dealing with mixtures of different sources.

A source separation problem with an incomplete and/or corrupted mixture is in fact a new problem that we introduce and address in this work, which, to our best knowledge, has not been properly solved by any of the existing source separation approaches in the literature, except a naive way: sequentially performing audio inpainting followed by source separation on the reconstructed mixture. The latter sequential approach can be quite suboptimal since neither of these two tasks use all of the information efficiently. The problem of jointly performing the two tasks is for the first time addressed by our proposed approach, which can recover the signal in a way that is more consistent with the multiple source nature of the corrupted mixture while simultaneously estimating the individual sources.

The global setup of our modeling to handle joint audio declipping and source separation is the same as the one for declipping in Section IV-A, except that $J > 1$ sources are considered instead of just one. In order to assess the performance of declipping and source separation using the proposed algorithm, 5 different music mixtures[5], each composed of 3 sources (bass, drums and vocals), are considered under 3 different clipping conditions. For each mixture with a maximum magnitude of 1 in time domain, 3 clipping levels at the thresholds of 0.2 (heavy clipping), 0.5 (moderate clipping) and 0.8 (light clipping) are considered, resulting in a total of 15 mixtures with different clipping levels. Each mixture is reconstructed by joint declipping and source separation, sequential declipping and source separation and only source separation ignoring the clipping artefacts. The proposed GEM algorithm (run for 100 iterations) has been used for all the reconstructions[6] with $K = 15$ components. Inline with [33] and so as to inject some information about the sources to be separated, the sources in the mixtures are artificially silenced during a percentage of the total time, and the corresponding indices in $\mathbf{H}$ are set to zero so as to inject this information into the modeling. An example of the activation periods of the sources and corresponding indices set to zero in $\mathbf{H}$ during NTF model estimation are shown in Figure 3. Similarly $\mathbf{Q}$ is simply chosen as a $J \times K$ matrix with a single 1 on each column and zeros everywhere else, to describe the assignment

[5]The mixtures are taken from the "professionally produced music recordings" task dataset of SiSEC 2015 source separation evaluation campaign (https://sisec.inria.fr/sisec-2015/).

[6]For declipping only, the algorithm is used with a single source (as in Sec. IV-A), and for source separation only, the algorithm is used with the observed support set being the entire time axis.

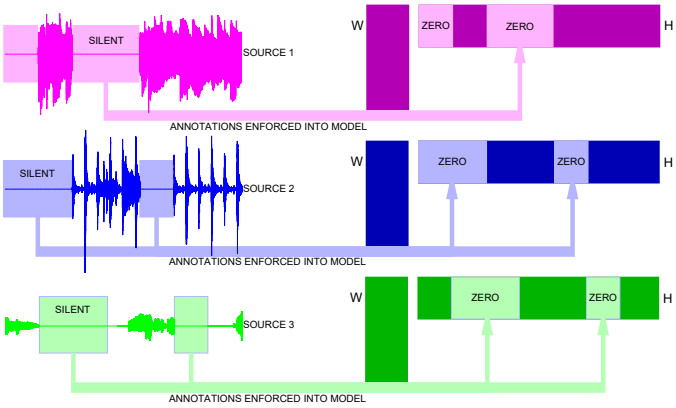


Fig. 3: Depiction of the experiment set-up for the injection of information on different source characteristics. The information on the known silent time durations of each source (represented in different colors) is directly utilized by setting the corresponding coefficients in $\mathbf{H}$ to 0. Note that the matrices $\mathbf{H}$ and $\mathbf{W}$ are formed by concatenating the Hs for each source and Ws for each source (depicted above) along the component dimension respectively.

of the components to the sources.

It should be noted that, the sequential reconstruction as well as performing only the source separation in this experiment could also have been performed with other existing methods from the literature. However, we have opted for using the same algorithm for each recovery scenario so as to clearly observe the difference due to jointly treating the two problems, rather than other differences in the reconstruction algorithms. Furthermore, as it is demonstrated that the performance of our algorithm for declipping is on par with the state of the art algorithms in Section IV-A, we find this comparison still very relevant.

The results of the simulations can be seen in Figure 4. Signal to noise ratio on the clipped support (SNR_m) computed as in (32) for the declipped mixture is shown to demonstrate the declipping performance while signal to distortion ratio (SDR) as described in [39] is shown to demonstrate the source separation performance.

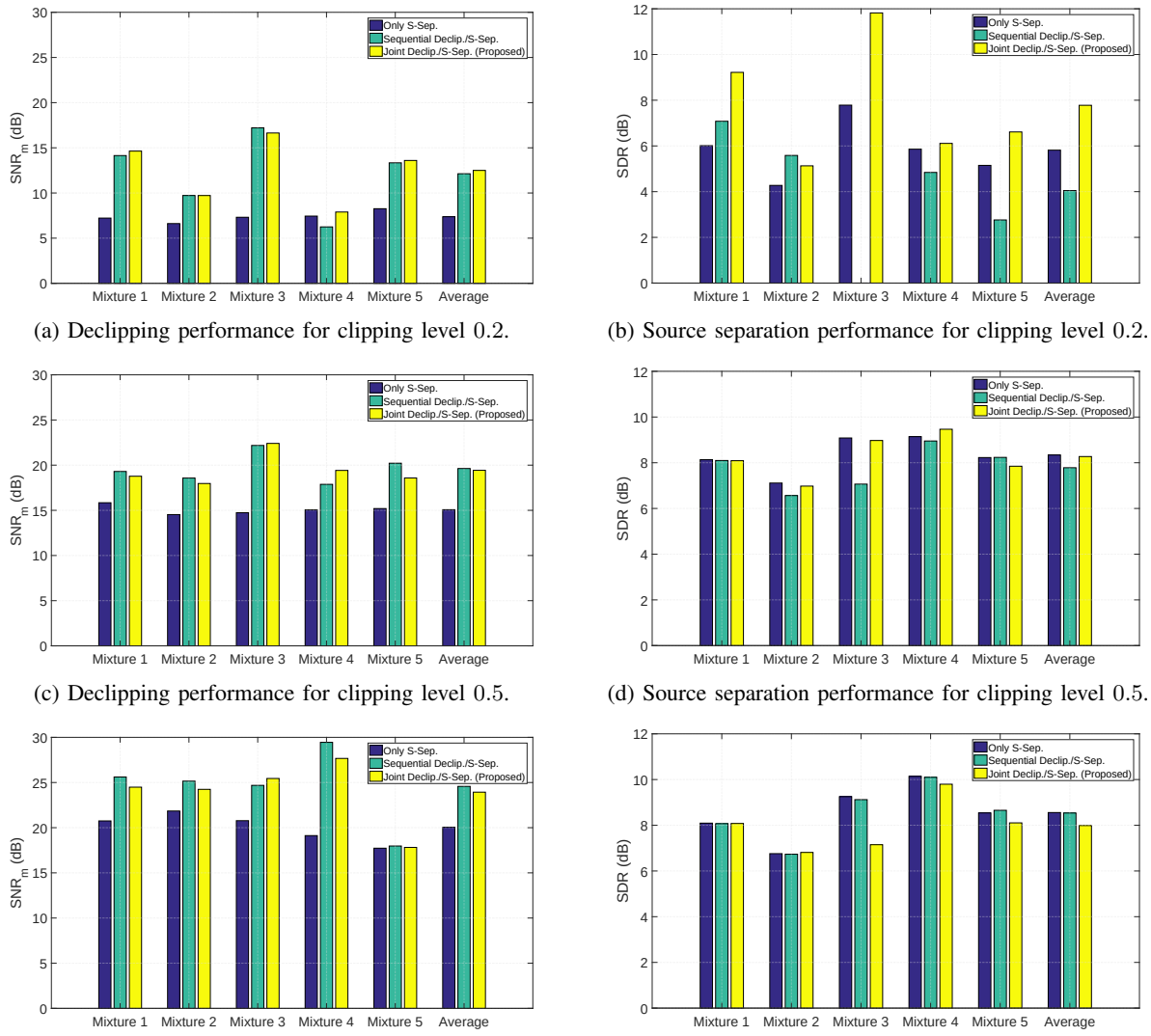
The results in Figure 4 show that when the clipping is severe, joint approach is almost always preferable since it provides improvement on both the quality of the mixture and the quality of the separated sources with respect to source separation without declipping. This is as opposed to the sequential approach which provides comparable quality improvement in the mixture at the expense of the performance in source separation. In fact, for heavy clipping the declipping in sequential approach often reduced the performance of source separation noticeably with respect to separation without declipping. As the clipping gets lighter, the performance of sequential method approaches to that of joint method, and finally performs slightly better for light clipping. The joint optimization, however, still has few drawbacks which could be improved upon. The declipping in the sequential approach is performed with $K = 15$ components without any restrictions whereas the joint optimization is performed with the additional limitation that each source uses 5 components *independently*. Hence it is not possible that two sources share a common component in the joint optimization. This can be overcome by devising better methods to inject the prior information regarding the sources, see, *e.g.*, [23]. It should be also noted that the sequential optimization is approximately twice as fast as joint optimization due to handling much less complicated problems in either steps of the sequential processing. The fact that the Wiener filtering stage is independent for each window and can be parallelized to provide significant speed improvements, can be helpful to overcome this problem in the future.

C. Compressive Sampling Recovery

Compressive sampling [29] is the theory and application of (often) randomly subsampling a signal that is known to be compressible (*e.g.*, with sparse or low rank representations) in an *incoherent* domain and making sense of the random samples by using the prior information of compressibility. As our algorithm is well fitted for time domain audio inverse problems, the reconstruction of the randomly sampled audio signals is another field of application for which it can be useful. Even though all the model-based signal estimations rely on compressibility of signals, the differentiating factor of compressive sampling comes from the fact that the compact representation of the signal is in an incoherent (in layman terms, very different or opposite) domain to the sampling domain. As an example, frequency domain and time domain are two domains which are maximally incoherent, *i.e.*, an impulse (maximally compact) signal in one is a uniform energy (maximally distributed) signal in the other.

Looking from the compressive sampling perspective, the compressible characteristics of the audio signals exploited by our algorithm are two fold: i) the significant reduction of the probability space of the possible solutions given the known samples, through the maximum likelihood estimate ii) the further reduction of the possible solutions through the low rank modeling of the NMF/NTF representation in the STFT domain. This application can in fact be seen as another instance of audio inpainting, however we have investigated it separately as the random subsampling changes the characteristic of the problem with respect to the other more typical audio inpainting problems such as audio declipping. It must be also noted that, this application is more than mere interpolation from irregular samples, as the reconstruction model enforces dimensionality reduction in an incoherent domain, fitting well into the compressive sensing paradigm.

In order to demonstrate the ability of the proposed approach to reconstruct randomly subsampled signals, we have randomly subsampled a typical music signal of 4 seconds at different average rates (percentage of retained samples at $2, 4, 8, 16, 32$), and then reconstructed with our algorithm in a similar fashion to the experiments in Section IV-A, but without any clipping constraints (hence $J = 1$ and $\sigma^2_{a,fn} = \sigma^2_{b,jfn} = 0$, $\forall j, f, n$). The reconstruction is performed with different number of components ($K = 2, 8, 24, 32, 48, 72$) in order to observe the sensitivity of the results to the parameter K. In



(a) Declipping performance for clipping level 0.2.

(b) Source separation performance for clipping level 0.2.

(c) Declipping performance for clipping level 0.5.

(d) Source separation performance for clipping level 0.5.

(e) Declipping performance for clipping level 0.8.

(f) Source separation performance for clipping level 0.8.

Fig. 4: The declipping and source separation performance of joint optimization compared to sequential.

order to provide a reference for the reconstruction capability of our algorithm, the results with shape preserving piecewise cubic interpolation are also provided[7].

The reconstruction results can be observed in Figure 5. The first thing to notice is that the reconstruction results with the proposed algorithm (solid lines) are significantly better than the results with simple interpolation (dashed lines) as expected. Another noticeable behaviour in the results is that once the number of components, K, is sufficiently large, the reconstruction performance does not seem to suffer. This behaviour is unlike what we have observed for other problems such as declipping, for which the choice of number of components is an important factor for obtaining best performance. Looking more closely to the estimated NMF components, we have seen that the maximum likelihood estimate combined with random sampling already provided a strong prior for signal estimation and the benefit from low rank model was minimal in this case. Hence, as long as the number of components are chosen sufficiently large, the accuracy of estimated variances, $\mathbf{V}$, are effectively independent of K.

[7]For the interpolation, the *interp1()* function of Matlab 2016a is used with *phcip* method, which gave the best results among the available interpolation methods.

D. Compressive sampling-based informed source separation

Informed source separation (ISS) [7], [30] is a variant of source separation that is in fact a source compression problem assuming that the mixture is known. The ISS problem can be defined as the problem of encoding multiple audio sources to create a bitstream (also called a *side-information*) so that the audio from the sources can be recovered given the bitstream and the mixture of the sources. The main difference of ISS from joint compression of multiple audio signals is the assumption that the mixture is available at both encoding and decoding stages. Several ISS methods were proposed [7], [30], [40] including those based on the NTF modeling [7], [30]. In all these approaches the encoding stage is usually more complex and computationally expensive than the decoding stage. The framework proposed in this work can be used

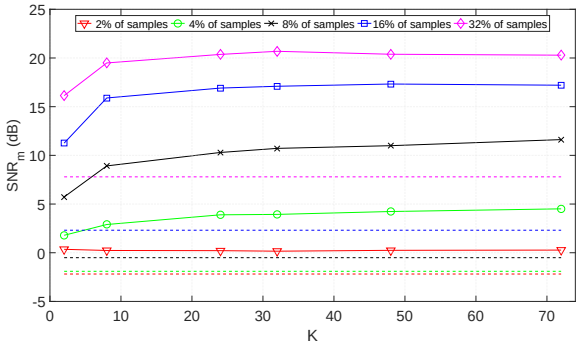
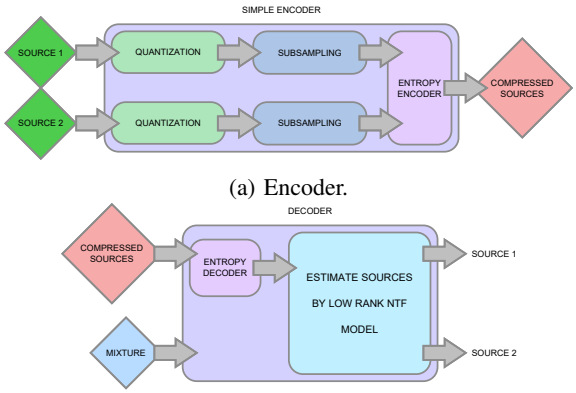


Fig. 5: The reconstruction performance measured in terms of SNR_m of a 4s long music signal from its random samples. The reconstruction results with our proposed algorithm (solid lines) are shown for different percentage of samples and different number of components, K, used in our approach. The results with shape preserving piecewise cubic interpolation are also shown for comparison (dashed lines), with the colors indicating corresponding percentage of samples.



(a) Encoder.

(b) Decoder.

Fig. 6: The encoding and decoding processes for the compressive sensing-based informed source separation.

to realize a new variant of ISS, where the computational complexity is moved from the encoder to the decoder side. To our best knowledge, this is another application that is realized for the first time with our proposed algorithm. This feat is accomplished by reducing the encoder to simply subsampling the sources in a random and independent fashion and quantizing the samples. The proposed algorithm can then be used to recover the sources at the decoder side given the encoded samples and the mixture, similar to the case of compressive sampling recovery (in fact this can be seen as more practical use of compressive sampling recovery in audio). This new approach, which we call *compressive sampling-based ISS (CS-ISS)*, is inline with both the compressive sampling [29] paradigm, since the sampling is random and in a sufficiently incoherent domain, and with the distributed source/video coding [41], [42], since the posterior source dependencies (the sources are highly correlated *a posteriori* given the mixture) and the source structure are exploited only at the decoding stage, thus allowing the complexity shift. The CS-ISS also allows independent structures between the encoder and the decoder, *i.e.*, the decoder algorithm can be modified without the need to change the encoder and the encoded bitstream. More precisely, by that we mean that given a bitstream a totally different source recovery algorithm (*e.g.*, based on social sparsity) may be developed and applied for decoding.

A summary of our CS-ISS scheme is shown in Figure 6. In order to assess the performance of our approach, three ($J = 3$) 11-second long sources of a music recording are encoded and then decoded using the proposed CS-ISS with different levels of quantization (16 bits, 11 bits, 6 bits and 1 bit) and different raw sampling bitrates[8] per source (0.64, 1.28, 2.56, 5.12 and 10.24 kbps/source). Since uniform quantization is used, the noise variance in time domain is $\sigma^2 = \Delta^2/12$ where Δ is the quantization step size. Hence $\sigma^2_{b,jfn} = \omega_f^2\Delta^2/12$, where ω_f^2 are the framing (or STFT) window coefficients. The mixture is available in entirety at the decoder, therefore the noise variance of the mixture is zero ($\sigma^2_{a,fn} = 0$). It is assumed that the random sampling pattern is *pre-defined* and known during both encoding and decoding. The quantized samples are truncated and compressed using an arithmetic encoder with a zero mean Gaussian distribution assumption. At the decoder side, following the arithmetic decoder, the sources are decoded from the quantized samples using 50 iterations of the GEM algorithm with the number of components fixed at $K = 18$, *i.e.* in average 6 components per source. The quality of the reconstructed samples is measured with SDR as described in [39]. The resulting encoded bitrates and SDR of decoded signals are presented in Table I along with the percentage of the encoded samples in parentheses. Note that the compressed rates in Table I differ from the corresponding raw bitrates due to the variable performance of the entropy coding stage, which is expected.

The performance of CS-ISS is compared to a classical ISS approach with a more complicated encoder and a simpler decoder presented in [30], as well as much better performing coding-based approach proposed in [7]. Both the classical ISS and coding-based ISS algorithms are used with NTF model quantization and encoding in a similar fashion as in the experiments described by [7], *i.e.*, NTF coefficients are uniformly quantized in logarithmic domain, quantization step sizes of different NTF matrices are computed using equations (31)-(33) from [7] and the indices are encoded using an arithmetic coder based on a two-state Gaussian mixture model (GMM) (see Fig. 5 of [7]). The approach is evaluated for different quantization step sizes and different numbers of NTF components, *i.e.*, $\Delta = 2^{-2}, 2^{-1.5}, 2^{-1}, \ldots, 2^4$ and $K = 4, 6, \ldots, 30$. The results are generated with 250 iterations of model update. The performance of CS-ISS and the earlier approaches are shown in Figure 7 in which CS-ISS clearly outperforms the classical ISS approach and is on par with coding-based ISS approach, even though both of these approaches can use an optimized number of components as opposed to our decoder which uses a fixed number of components (the encoder is very simple and does not compute or transmit this value). The performance

[8]The raw sampling bitrate is defined as the bitrate before the entropy encoding step.

Bits per Sample	Raw rate (kbps / source)				
	0.64	**1.28**	**2.56**	**5.12**	**10.24**
	Compressed Rate / SDR (% of Samples Kept)				
16 bits	0.50 / -1.64 dB (0.25%)	1.00 / 4.28 dB (0.50%)	2.00 / 9.54 dB (1.00%)	4.01 / 16.17 dB (2.00%)	8.00 / 21.87 dB (4.00%)
11 bits	0.43 / 1.30 dB (0.36%)	0.87 / 6.54 dB (0.73%)	1.75 / 13.30 dB (1.45%)	**3.50 / 19.47 dB** (2.91%)	**7.00 / 24.66 dB** (5.82%)
6 bits	**0.27 / 4.17 dB** (0.67%)	**0.54 / 7.62 dB** (1.33%)	**1.08 / 12.09 dB** (2.67%)	2.18 / 14.55 dB (5.33%)	4.37 / 16.55 dB (10.67%)
1 bit	0.64 / -5.06 dB (4.00%)	1.28 / -2.57 dB (8.00%)	2.56 / 1.08 dB (16.00%)	5.12 / 1.59 dB (32.00%)	10.24 / 1.56 dB (64.00%)

TABLE I: The final bitrates (in kbps per source) after the entropy coding stage of CS-ISS with corresponding SDR (in dBs) for different (uniform) quantization levels and different raw bitrates before entropy coding. The percentage of the samples kept is also provided for each case in parentheses. Results corresponding to the best rate-distortion compromise are in bold.

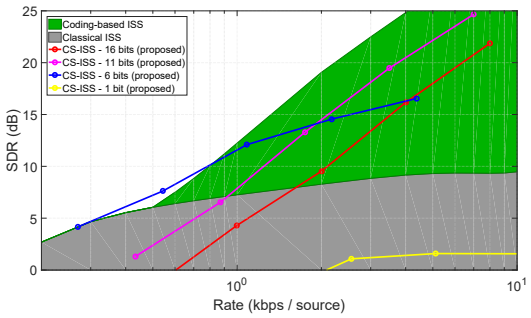


Fig. 7: The rate-distortion performance of CS-ISS using different quantization levels of the encoded samples. The performance of the ISS algorithm from [30] and the coding-based ISS algorithm from [7] are also shown for comparison.

difference with classical ISS is due to the high efficiency achieved by the CS-ISS decoder thanks to the incoherency of random sampled time domain and of maximum likelihood estimation along with low rank NTF model. Also, the classical ISS approach [30] is unable to perform beyond an SDR of 10 dBs due to the lack of additional information about STFT phase as explained in [7]. The results indicate that the rate distortion performance exhibits a similar behaviour as to the coding-based ISS algorithm. It should be reminded that the proposed approach distinguishes itself by its low complexity encoder and hence can still be advantageous against other ISS approaches with better or seemingly equivalent rate distortion performance.

The performance of CS-ISS in Table I and Figure 7 indicates that different levels of quantization may be preferable in different rates. Even though neither 16 bits nor 1 bit quantization seem well performing, the performance indicates that 16 bits quantization may be superior to other schemes when a much higher bitrate is available. Coarser quantization such as 1 bit, on the other hand, had very poor performance in the experiments. The choice of quantization can be performed in the encoder with a simple look up table as a reference. One must also note that even though the encoder in CS-ISS is very simple, the proposed decoder is significantly high complexity, typically higher than the encoders of traditional ISS methods. However, this can also be overcome by exploiting the independence of Wiener filtering among the frames in the proposed decoder with parallel processing, *e.g.*, using GPUs.

V. Conclusions

In this paper, we have presented a novel approach for time domain signal estimation in the maximum likelihood manner. It relies on the low rank NTF modeling of the power spectrum of the signal and can be applied to many types of problems that were not previously solved using the NMF/NTF model. The proposed algorithm is demonstrated to be very effective for several audio inverse problems while providing multiple advantages compared to other existing methods. For the audio declipping problem, clipped sections of music and speech signals are restored using the proposed approach as well as state of the art methods, and the proposed algorithm is shown to be highly competitive while providing complementary advantages such as naturally handling noise and quantization artefacts and easily incorporating various types of constraints. For audio source separation and mixture declipping, the proposed algorithm is shown to be capable of jointly solving these two separate problems which was not possible with any other method in the literature. Joint handling of these problems is also demonstrated to be more effective than sequentially approaching each problem in case of severe distortions. The proposed algorithm is also shown to be highly effective for the reconstruction of randomly subsampled signals such as in the case of compressive sampling approaches. This advantage of our algorithm is further utilised for the problem of informed source separation, to create a compression scheme which uses the principles of compressive sampling and distributed coding. For this application, the proposed algorithm is not only shown to achieve compression performance equivalent to that of the state of the art, but also shown to have unique advantages, specifically having a very simple encoder as well as the decoding stage being independent of the encoding stage.

The NMF and NTF representations are gaining a lot of popularity in signal modelling community and we see the algorithm presented in this paper to be a step towards the application of these models to a wider class of signal estimation problems. Even though the provided examples in this paper are all audio inverse problems, the proposed algorithm is by no means limited to audio applications. It could be used in any application for which a low rank NMF/NTF model is an accurate representation for the power spectrum.

We consider several improvements and extensions to the proposed algorithm as future work. An extension to multichannel audio is an interesting step for dealing with real world audio problems. Furthermore, adapting the proposed algorithm

for imaging problems with multiple additive components, such as imaging through transparent and reflective surfaces, is another intriguing direction.

VI. Acknowledgment

The authors would like to thank Kai Siedenburg and Matthieu Kowalski for kindly sharing numerical results from [28], and Srđan Kitić for providing the results of the algorithm from [27] on the corresponding dataset.

References

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

[2] A. Cichocki, R. Zdunek, and S. Amari, "Nonnegative matrix and tensor factorization," *IEEE Signal Processing Magazine*, pp. 142–145, 2008.

[3] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[4] C. Févotte, N. Bertin, and J. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, Mar. 2009.

[5] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 3, pp. 550–563, Mar. 2010.

[6] J. Nikunen and T. Virtanen, "Object-based audio coding using non-negative matrix factorization for the spectrogram representation," in *128th Audio Engineering Society Convention (AES 2010)*, London, UK, May 2010.

[7] A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, "Coding-based informed source separation: Nonnegative tensor factorization approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 8, pp. 1699–1712, Aug. 2013.

[8] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct. 2003.

[9] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 528–537, Mar. 2010.

[10] J. L. Roux, H. Kameoka, N. Ono, A. de Cheveigné, and S. Sagayama, "Computational auditory induction as a missing-data model-fitting problem with Bregman divergence," *Speech Communication*, vol. 53, no. 5, pp. 658–676, May-June 2011.

[11] P. Smaragdis, R. Bhiksha, and S. Madhusudana, "Missing data imputation for time-frequency representations of audio signals," *Journal of signal processing systems*, vol. 65, no. 3, pp. 361–370, 2011.

[12] U. Simsekli, A. T. Cemgil, and Y. K. Yilmaz, "Score guided audio restoration via generalised coupled tensor factorisation," in *International Conference on Acoustics Speech and Signal Processing (ICASSP'12)*, 2012, pp. 5369 – 5372.

[13] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, "Audio inpainting," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 3, pp. 922 – 932, 2012.

[14] D. Griffin and J. S. Lim, "Signal reconstruction from short-time Fourier transform magnitude," *IEEE Transactions of Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[15] R. Badeau and M. D. Plumbley, "Multichannel high-resolution NMF for modeling convolutive mixtures of non-stationary signals in the time-frequency domain," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 11, pp. 1670–1680, 2014.

[16] R. Badeau and A. Ozerov, "Multiplicative updates for modeling mixtures of non-stationary signals in the time-frequency domain," in *Proc. 21st European Signal Processing Conference (EUSIPCO)*, Marrakech, Morocco, Sep. 2013.

[17] A. Dempster, N. Laird, and D. Rubin., "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, pp. 1–38, 1977.

[18] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice Hall, 1993.

[19] Ç. Bilen, A. Ozerov, and P. Pérez, "Audio declipping via nonnegative matrix factorization," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 2015.

[20] ——, "Joint audio inpainting and source separation," in *The 12th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA 2015)*, August 2015.

[21] ——, "Compressive sampling-based informed source separation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 2015.

[22] ——, "Audio inpainting, source separation, audio compression. all with a unified framework based on ntf model," in *MissData 2015*, 2015.

[23] ——, "Automatic allocation of NTF components for user-guided audio source separation," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 484–488.

[24] A. Ozerov, Ç. Bilen, and P. Pérez, "Multichannel audio declipping," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 659–663.

[25] S. Kitić, L. Jacques, N. Madhu, M. Hopwood, A. Spriet, and C. D. Vleeschauwer, "Consistent iterative hard thresholding for signal declipping," in *ICASSP - The 38th International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 2013.

[26] S. Kitić, N. Bertin, and R. Gribonval, "Audio declipping by cosparse hard thresholding," in *iTwist - 2nd international - Traveling Workshop on Interactions betweenSparse models and Technology*, Namur, Belgium, August 2014.

[27] ——, "Sparsity and cosparsity for audio declipping: a flexible non-convex approach," in *The 12th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA 2015)*, August, 2015.

[28] K. Siedenburg, M. Kowalski, and M. Dörfler, "Audio declipping with social sparsity," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 1577–1581.

[29] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, pp. 21–30, 2008.

[30] A. Liutkus, J. Pinel, R. Badeau, L. Girin, and G. Richard, "Informed source separation through spectrogram coding and data embedding," *Signal Processing*, vol. 92, no. 8, pp. 1937–1949, 2012.

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

[32] R. Bro, "Parafac. tutorial and applications," *Chemometrics and intelligent laboratory systems*, vol. 38, no. 2, pp. 149–171, 1997.

[33] A. Ozerov, C. Févotte, R. Blouet, and J.-L. Durrieu, "Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'11)*, Prague, May 2011, pp. 257–260.

[34] J. Le Roux, F. Weninger, and J. R. Hershey, "Sparse NMF? half-baked or well done?" Mitsubishi Electric Research Laboratories, Tech. Rep., 2015.

[35] A. Janssen, R. Veldhuis, and L. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 317–330, 1986.

[36] S. J. Godsill and P. J. Rayner, "A Bayesian approach to the restoration of degraded audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 4, pp. 267–278, 1995.

[37] W. Etter, "Restoration of a discrete-time signal segment by interpolation based on the left-sided and right-sided autoregressive parameters," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1124–1135, 1996.

[38] A. Dahimene, M. Noureddine, and A. Azrar, "A simple algorithm for the restoration of clipped speech signal," *Informatica*, vol. 32, no. 2, 2008.

[39] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.

[40] M. Parvaix and L. Girin, "Informed source separation of linear instantaneous under-determined audio mixtures by source index embedding," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 6, pp. 1721 – 1733, 2011.

[41] Z. Xiong, A. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, September 2004.

[42] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71 – 83, January 2005.