



HAL
open science

Phase recovery with Bregman divergences for audio source separation

Paul Magron, Pierre-Hugo Vial, Thomas Oberlin, Cédric Févotte

► **To cite this version:**

Paul Magron, Pierre-Hugo Vial, Thomas Oberlin, Cédric Févotte. Phase recovery with Bregman divergences for audio source separation. 2020. hal-03049800v1

HAL Id: hal-03049800

<https://hal.science/hal-03049800v1>

Preprint submitted on 10 Dec 2020 (v1), last revised 9 Feb 2021 (v2)

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

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

Phase recovery with Bregman divergences for audio source separation*

Paul Magron[†], Pierre-Hugo Vial[†], Thomas Oberlin[‡], Cédric Févotte[†]

Abstract

Time-frequency audio source separation is usually achieved by estimating the short-time Fourier transform (STFT) magnitude of each source, and then applying a phase recovery algorithm to retrieve time-domain signals. In particular, the multiple input spectrogram inversion (MISI) algorithm has shown good performance in several recent works. This algorithm minimizes a quadratic reconstruction error between magnitude spectrograms. However, this loss does not properly account for some perceptual properties of audio, and alternative discrepancy measures such as beta-divergences have been preferred in many settings. In this paper, we propose to reformulate phase recovery in audio source separation as a minimization problem involving Bregman divergences. To optimize the resulting objective, we derive a projected gradient descent algorithm. Experiments conducted on a speech enhancement task show that this approach outperforms MISI for several alternative losses, which highlights their relevance for audio source separation applications.

Keywords— Phase recovery, Bregman divergences, projected gradient descent, audio source separation, speech enhancement.

1 Introduction

Audio source separation [1] consists in extracting the underlying *sources* that add up to form an observable audio *mixture*. This task finds applications in many areas such as speech enhancement and recognition [2] or musical signal processing [3]. State-of-the-art approaches for source separation consist in using a deep neural network (DNN) or nonnegative matrix factorization (NMF) to estimate a nonnegative mask that is applied to a time-frequency (TF) representation of the audio mixture, such as the short-time Fourier transform (STFT) [4]. Recent works such as [5, 6] operate in the time domain directly, but TF approaches remain interesting since they allow to better exploit the structure of sound [7].

Applying a nonnegative mask to the mixture’s STFT results in assigning its phase to each isolated source. Even though this practice is common and yields satisfactory results, it is well established [8] that when sources overlap in the TF domain, using the mixture’s phase induces residual interference and artifacts in the estimates. With the advent of deep learning, magnitudes can nowadays be estimated with a high accuracy, which outlines the need for more advanced phase recovery algorithms [9]. Consequently, a significant research effort has been put on phase recovery in DNN-based source separation, whether phase recovery algorithms are applied as a post-processing [8] or integrated within end-to-end systems for time-domain separation [10, 11, 12].

Among the variety of phase recovery techniques, the multiple input spectrogram inversion (MISI) algorithm [13] is particularly popular. This iterative procedure consists in retrieving time-domain sources from their STFT magnitudes while respecting a mixing constraint: the estimates must add up to the mixture. This algorithm exhibits a good performance in source separation when combined with DNNs [10, 11]. However, MISI suffers from one limitation. Indeed, it is derived as a solution to an optimization problem that involves the quadratic loss, which is not the best-suited metric for evaluating discrepancies in the TF domain. For instance, it does not properly account for the large dynamic range of audio signals [14].

In this work, we consider phase recovery in audio source separation as an optimization problem involving alternative divergences which are more appropriate for audio processing. We consider general Bregman divergences, a family of loss functions which encompasses the β -divergence [15] and some of its well-known special cases, the Kullback-Leibler (KL) and Itakura-Saito (IS) divergences. These divergences are acknowledged for their superior performance in audio spectral decomposition applications such as NMF-based source separation [16]. In a previous work [17], we addressed phase recovery with the Bregman divergences in a single-source setting. Here, we propose to extend this approach to a single-channel and multiple-sources framework, where the mixture’s information can be exploited. To optimize the resulting objective, we derive a projected gradient algorithm [18].

*This work is supported by the European Research Council (ERC FACTORY-CoG-6681839).

[†]IRIT, Université de Toulouse, CNRS, Toulouse, France (e-mail: firstname.lastname@irit.fr).

[‡]ISAE-SUPAERO, Université de Toulouse, France (e-mail: firstname.lastname@isae-supaero.fr).

We experimentally assess the potential of our approach for a speech enhancement task. Our results show that this method outperforms MISI for several Bregman divergences.

The rest of this paper is structured as follows. Section 2 presents the related work. In Section 3 we derive the proposed algorithm. Section 4 presents the experimental results. Finally, Section 5 draws some concluding remarks.

Mathematical notations:

- \mathbf{A} (capital, bold font): matrix.
- \mathbf{s} (lower case, bold font): vector.
- $\text{diag}(\mathbf{u}) \in \mathbb{C}^{K \times K}$: diagonal matrix whose entries are the elements of $\mathbf{u} \in \mathbb{C}^K$.
- z (regular): scalar.
- $|\cdot|, \angle(\cdot)$: magnitude and complex angle, respectively.
- $\mathbf{s}^\top, \mathbf{s}^H$: transpose and Hermitian transpose, respectively.
- $\Re(\cdot), \Im(\cdot)$: real and imaginary part functions, respectively.
- $\|\cdot\|_2$: Euclidean norm.
- $\odot, \div, (\cdot)^d$: element-wise matrix or vector multiplication, division, and power, respectively.

2 Related work

In this section, we present the necessary background upon which our work builds. We describe the baseline phase recovery problem (Section 2.1), its extension to multiple sources (Section 2.2), and its formulation using the Bregman divergences (Section 2.3).

2.1 Phase recovery

Phase recovery is commonly formulated as the following problem:

$$\min_{\mathbf{s} \in \mathbb{R}^L} \|\mathbf{r} - |\mathbf{A}\mathbf{s}|^d\|_2^2, \quad (1)$$

where $\mathbf{r} \in \mathbb{R}_+^K$ are nonnegative measurements, usually an STFT magnitude ($d = 1$) or power ($d = 2$) spectrogram, and $\mathbf{A} \in \mathbb{C}^{K \times L}$ is the matrix that encodes the STFT. In the seminal work [19], the authors address problem (1) with $d = 1$. Starting from an initial guess $\mathbf{s}^{(0)}$, they propose the following update rule:

$$\mathbf{s}^{(t+1)} = \mathbf{A}^\dagger \left(\mathbf{r} \odot \frac{\mathbf{A}\mathbf{s}^{(t)}}{|\mathbf{A}\mathbf{s}^{(t)}|} \right) \quad (2)$$

where \mathbf{A}^\dagger is the Moore-Penrose pseudo-inverse of \mathbf{A} defined as $\mathbf{A}^\dagger = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$, which encodes the inverse STFT. This iterative scheme, known as the Griffin-Lim (GL) algorithm, is proved to converge to a critical point of the quadratic loss in (1) [19], and can also be obtained by majorization-minimization [20] or using a gradient descent scheme [17]. Improvements of this algorithm notably include accelerated [21] and real-time purposed versions [22].

2.2 Multiple input spectrogram inversion (MISI)

The GL algorithm has been extended to handle multiple sources in a source separation framework [13]. Given an observed mixture $\mathbf{x} \in \mathbb{R}^L$ of C sources $\mathbf{s}_c \in \mathbb{R}^L$, whose target nonnegative TF measurements are \mathbf{r}_c , this problem can be formulated as [23]:

$$\min_{\{\mathbf{s}_c \in \mathbb{R}^L\}_{c=1}^C} \sum_{c=1}^C \|\mathbf{r}_c - |\mathbf{A}\mathbf{s}_c|^d\|_2^2 \text{ s.t. } \sum_{c=1}^C \mathbf{s}_c = \mathbf{x}. \quad (3)$$

The multiple input spectrogram inversion (MISI) algorithm addresses (3) when $d = 1$ and consists of the following updates:

$$\forall c, \mathbf{y}_c^{(t)} = \mathbf{A}^\dagger \left(\mathbf{r} \odot \frac{\mathbf{A}\mathbf{s}_c^{(t)}}{|\mathbf{A}\mathbf{s}_c^{(t)}|} \right) \quad (4)$$

$$\forall c, \mathbf{s}_c^{(t+1)} = \mathbf{y}_c^{(t)} + \frac{1}{C} \left(\mathbf{x} - \sum_{i=1}^C \mathbf{y}_i^{(t)} \right) \quad (5)$$

In a nutshell, this algorithm consists in performing the GL update (2) for each source individually, and then distributing the resulting mixing error onto those estimates to yield a set of signals $\{\mathbf{s}_c\}$ that add up to the mixture. The MISI algorithm has been introduced heuristically in [13]. In [23], it was derived using a majorization-minimization strategy, which proved its convergence.

2.3 Phase recovery with the Bregman divergence

In [17], we proposed to replace the quadratic loss in problem (1) with Bregman divergences, which encompass the β -divergence [15] and its special cases, the KL and IS divergences. A Bregman divergence \mathcal{D}_ψ is defined from a strictly-convex, continuously-differentiable generating function ψ (with derivative ψ') as follows:

$$\mathcal{D}_\psi(\mathbf{r} \mid \mathbf{z}) = \sum_k [\psi(r_k) - \psi(z_k) - \psi'(z_k)(r_k - z_k)]. \quad (6)$$

Typical Bregman divergences with their generating function and derivative can be found, e.g., in [17] (see Table 1). Since the Bregman divergences are not symmetric in general, we considered the following two different problems, respectively termed “left” and “right”:

$$\min_{\mathbf{s} \in \mathbb{R}^L} \mathcal{D}_\psi(\mathbf{r} \mid |\mathbf{A}\mathbf{s}|^d), \quad (7)$$

$$\min_{\mathbf{s} \in \mathbb{R}^L} \mathcal{D}_\psi(|\mathbf{A}\mathbf{s}|^d \mid \mathbf{r}). \quad (8)$$

In [17] we derived two algorithms for solving both problems, based on gradient descent and alternating direction method of multipliers (ADMM).

3 Proposed method

3.1 Problem setting

We propose to extend our previous approach described in Section 2.3 to a single-channel source separation framework. Indeed, as described in Section 2.2, it is necessary to include the mixture information in the optimization problem so that the estimates add up to the mixture. We replace the loss in (3) with a Bregman divergence, as in (7), which yields the following optimization problem:

$$\min_{\{\mathbf{s}_c \in \mathbb{R}^L\}_{c=1}^C} \sum_{c=1}^C J_c(\mathbf{s}_c) \quad \text{s.t.} \quad \sum_{c=1}^C \mathbf{s}_c = \mathbf{x}, \quad (9)$$

where $J_c(\mathbf{s}_c) = \mathcal{D}_\psi(\mathbf{r}_c \mid |\mathbf{A}\mathbf{s}_c|^d)$ for the “right” problem and $J_c(\mathbf{s}_c) = \mathcal{D}_\psi(|\mathbf{A}\mathbf{s}_c|^d \mid \mathbf{r}_c)$ for its “left” counterpart.

3.2 Projected gradient descent

Similarly to [17], we propose a gradient descent algorithm to minimize the objective defined in (9). The set of signals whose sum is equal to the observed mixture \mathbf{x} , appearing in the constraint of (9), is convex. As such, we may use the projected gradient algorithm [18] which boils down to alternating the two following updates:

$$\forall c, \mathbf{y}_c^{(t)} = \mathbf{s}_c^{(t)} - \mu \nabla J_c(\mathbf{s}_c^{(t)}) \quad (10)$$

$$\forall c, \mathbf{s}_c^{(t+1)} = \mathbf{y}_c^{(t)} + \frac{1}{C} \left(\mathbf{x} - \sum_{i=1}^C \mathbf{y}_i^{(t)} \right) \quad (11)$$

where ∇J_c denotes the gradient of J_c with respect to \mathbf{s}_c and $\mu > 0$ is the gradient step size. In a nutshell, (10) performs a gradient descent, and, similarly to (5), (11) projects the auxiliary variables \mathbf{y}_c onto the set of estimates whose sum is equal to the mixture.

3.3 Derivation of the gradient

We derive hereafter the gradient of J_c . Using the chain rule [24], we have:

$$\nabla J_c(\mathbf{s}_c) = (\nabla |\mathbf{A}\mathbf{s}_c|^d)^\top \mathbf{z}_c, \quad (12)$$

where $\nabla |\mathbf{A}\mathbf{s}_c|^d$ denotes the Jacobian of the multivariate function $\mathbf{s}_c \rightarrow |\mathbf{A}\mathbf{s}_c|^d$ (the Jacobian being the extension of the gradient for multivariate functions, we may use the same notation ∇), and:

$$\text{for the “right” problem, } \mathbf{z}_c = \psi''(|\mathbf{A}\mathbf{s}_c|^d) \odot (|\mathbf{A}\mathbf{s}_c|^d - \mathbf{r}_c)$$

$$\text{for the “left” problem, } \mathbf{z}_c = \psi'(|\mathbf{A}\mathbf{s}_c|^d) - \psi'(\mathbf{r}_c)$$

where ψ' and ψ'' are applied entrywise. Now, let us note \mathbf{A}_r and \mathbf{A}_i the real and imaginary parts of \mathbf{A} , respectively. Using differentiation rules for element-wise matrix operations [24] and calculations similar to [17], we have:

$$\begin{aligned} \nabla |\mathbf{A}\mathbf{s}_c|^d &= \nabla ((\mathbf{A}_r \mathbf{s}_c)^2 + (\mathbf{A}_i \mathbf{s}_c)^2)^{\frac{d}{2}} \\ &= d \times \text{diag}(|\mathbf{A}\mathbf{s}_c|^{d-2}) (\text{diag}(\mathbf{A}_r \mathbf{s}_c) \mathbf{A}_r + \text{diag}(\mathbf{A}_i \mathbf{s}_c) \mathbf{A}_i). \end{aligned} \quad (13)$$

Algorithm 1: Phase recovery with the Bregman divergence for audio source separation: gradient descent.

1 **Inputs:** Measurements $\mathbf{R}_c \in \mathbb{R}_+^{M \times N}$, mixture $\mathbf{x} \in \mathbb{R}^L$, step size $\tilde{\mu} > 0$, Bregman divergence function ψ .

2 **Initialization:**

3 $\forall c, \mathbf{s}_c = \text{iSTFT}(\mathbf{R}_c^{1/d} \odot \frac{\text{STFT}(\mathbf{x})}{|\text{STFT}(\mathbf{x})|})$

4 **while** *stopping criteria not reached* **do**

5 $\forall c, \mathbf{S}_c = \text{STFT}(\mathbf{s}_c)$

6 **if** “right” **then**

7 $\mathbf{Z}_c = \psi''(|\mathbf{S}_c|^d) \odot (|\mathbf{S}_c|^d - \mathbf{R}_c)$

8 **else if** “left” **then**

9 $\mathbf{Z}_c = \psi'(|\mathbf{S}_c|^d) - \psi'(\mathbf{R}_c)$

10 $\forall c, \mathbf{g}_c = d \times \text{iSTFT}(\mathbf{S}_c \odot |\mathbf{S}_c|^{d-2} \odot \mathbf{Z}_c)$

11 $\forall c, \mathbf{y}_c = \mathbf{s}_c - \tilde{\mu} \mathbf{g}_c$

12 $\forall c, \mathbf{s}_c = \mathbf{y}_c + (\mathbf{x} - \sum_{i=1}^C \mathbf{y}_i) / C$

13 **end**

14 **Output:** $\{\mathbf{s}_c\}_{c=1}^C$

We now inject (13) in (12) and develop, which yields:

$$\nabla J_c(\mathbf{s}_c) = \mathbf{A}_r^\top \left(d \times \text{diag}(\mathbf{A}_r \mathbf{s}_c) \text{diag}(|\mathbf{A}_r \mathbf{s}_c|^{d-2}) \mathbf{z}_c \right) + \mathbf{A}_i^\top \left(d \times \text{diag}(\mathbf{A}_i \mathbf{s}_c) \text{diag}(|\mathbf{A}_i \mathbf{s}_c|^{d-2}) \mathbf{z}_c \right). \quad (14)$$

We remark that $\forall \mathbf{u}, \mathbf{v} \in \mathbb{C}^K$, $\text{diag}(\mathbf{u})\mathbf{v} = \mathbf{u} \odot \mathbf{v}$, so we further simplify this expression:

$$\nabla J_c(\mathbf{s}_c) = \mathbf{A}_r^\top \left(d \times (\mathbf{A}_r \mathbf{s}_c) \odot |\mathbf{A}_r \mathbf{s}_c|^{d-2} \odot \mathbf{z}_c \right) + \mathbf{A}_i^\top \left(d \times (\mathbf{A}_i \mathbf{s}_c) \odot |\mathbf{A}_i \mathbf{s}_c|^{d-2} \odot \mathbf{z}_c \right). \quad (15)$$

Finally, we remark that $\forall \mathbf{u} \in \mathbb{C}^K$, $\Re(\mathbf{A}^H \mathbf{u}) = \mathbf{A}_r^\top \Re(\mathbf{u}) + \mathbf{A}_i^\top \Im(\mathbf{u})$, thus we can rewrite the gradient (15) as:¹

$$\nabla J_c(\mathbf{s}_c) = d \times \Re \left(\mathbf{A}^H ((\mathbf{A}_r \mathbf{s}_c) \odot |\mathbf{A}_r \mathbf{s}_c|^{d-2} \odot \mathbf{z}_c) \right). \quad (16)$$

3.4 Implementation of the gradient update

It is common practice to use the same window for computing both the STFT and its inverse, up to a normalization constant b , which ensures perfect reconstruction for usual windows (e.g., Hann or Hamming) and overlap ratios (e.g., 50 or 75 %) [25]. In such a setting, $\mathbf{A}^H \mathbf{A} = b\mathbf{I}$, and thus $\mathbf{A}^\dagger = \frac{1}{b} \mathbf{A}^H$: consequently, \mathbf{A}^H encodes the inverse STFT up to this normalization constant.

Let us also point out that when processing audio signals, applying \mathbf{A}^H returns real-valued signals [17]. We can therefore ignore the extra real part in (16). The gradient update (10) then rewrites:

$$\forall c, \mathbf{y}_c^{(t)} = \mathbf{s}_c^{(t)} - \tilde{\mu} d \times \left(\mathbf{A}^\dagger ((\mathbf{A}_r \mathbf{s}_c) \odot |\mathbf{A}_r \mathbf{s}_c|^{d-2} \odot \mathbf{z}_c) \right), \quad (17)$$

where $\tilde{\mu} = \mu/b$ is the normalized step size, which we simply term “step size” in what follows.

Remark: When considering the quadratic loss (for which the “right” and “left” problems are equivalent) with $d = 1$ and step size $\tilde{\mu} = 1$, the gradient update (17) becomes equivalent to the MISI update (4). This outlines that our method generalizes MISI, as the latter can be seen as a particular case of the projected gradient descent algorithm.

3.5 Algorithm

The proposed algorithm consists of alternating the updates (17) and (11). A natural choice for obtaining initial source estimates consists in assigning the mixture’s phase to each source’s STFT, which is known as *amplitude masking* and is commonly employed to initialize MISI [10, 11, 13]:

$$\forall c, \mathbf{s}_c^{(0)} = \mathbf{A}^\dagger \left(\mathbf{r}_c^{1/d} \odot \frac{\mathbf{A}\mathbf{x}}{|\mathbf{A}\mathbf{x}|} \right). \quad (18)$$

The STFT matrix \mathbf{A} and its inverse are large structured matrices that allow for efficient implementations of matrix-vector products. In that setting, it is more customary to handle TF matrices of size $M \times N$, where M is the number of frequency channels and N the number of time frames, rather than vectors of size $K = MN$. As such, we provide in Algorithm 1 the pseudo-code for practical implementation of our method.

¹Note that this gradient is not defined when at least one entry of $\mathbf{A}_r \mathbf{s}_c$ is null for $d = 1$ and/or $\beta \leq 1$. However, in practice, we manipulate STFTs whose entries are all non-zero, which alleviates this potential issue.

4 Experiments

In this section, we assess the potential of Algorithm 1 for a speech enhancement task, that is, with $C = 2$ and where \mathbf{s}_1 and \mathbf{s}_2 correspond to the clean speech and noise, respectively. In the spirit of reproducible research, we will release the code related to those experiments along with the final version of the paper.

4.1 Protocol

Data. As acoustic material, we build a set of mixtures of clean speech and noise. The clean speech is obtained from the VoiceBank test set [26], from which we randomly select 100 utterances. The noise signals are obtained from the DEMAND dataset [27], from which we select noises from three real-world environments: a living room, a bus, and a public square. For each clean speech signal, we randomly select a noise excerpt cropped at the same length than that of the speech signal. We then mix the two signals at various input signal-to-noise ratios (SNRs) (10, 0, and -10 dB). All audio excerpts are single-channel and sampled at 16,000 Hz. The STFT is computed with a 1024 samples-long (64 ms) Hann window, no zero-padding, and 75% overlap. The dataset is split into two subsets of 50 mixtures: a *validation* set, on which the step size is tuned (see Section 4.2); and a *test* set, on which the proposed algorithm is compared to MISI.

Spectrogram estimation. In realistic scenarios, the nonnegative measurements \mathbf{r}_c are estimates of the magnitude or power spectrograms of the sources. To obtain such estimates, we use Open-Unmix [28], an open implementation of a three-layer BLSTM neural network, originally tailored for music source separation applications. This network has been adapted to a speech enhancement task. It was trained on our dataset, except using different speakers and noise environments, as described in [29]. We use the trained model available at [30]. This network is fed with the noisy mixtures and outputs an estimate for the clean speech and noise spectrograms, which serve as inputs to the phase retrieval methods.

Compared methods. We test the proposed projected gradient descent method described in Algorithm 1 in a variety of settings. We consider magnitude and power measurements ($d = 1$ or 2), “right” and “left” problems, and various values of β for the divergence ($\beta = 0$ to 2 with a step of 0.25). The step size is tuned on the validation set. As comparison baseline, we consider the MISI algorithm (which corresponds to our algorithm with $\beta = 2$, $d = 1$ and $\tilde{\mu} = 1$). Following traditional practice with MISI [10, 11], all algorithms are run with 5 iterations. In order to evaluate the speech enhancement quality, we compute the signal-to-distortion ratio (SDR) between the true clean speech \mathbf{s}_1^* and its estimate \mathbf{s}_1 (higher is better):

$$\text{SDR}(\mathbf{s}_1^*, \mathbf{s}_1) = 20 \log_{10} \frac{\|\mathbf{s}_1^*\|_2}{\|\mathbf{s}_1^* - \mathbf{s}_1\|_2}. \quad (19)$$

For more clarity, we will present the SDR improvement (SDRi) of a method (whether MISI or Algorithm 1) over initialization.

4.2 Influence of the step size

First, we study the impact of the step size on the performance of the proposed algorithm using the validation set. The mean SDRi on this subset is presented in Figure 1 in the “right” setting, but similar conclusions can be drawn in the “left” setting. For $d = 1$, we remark that the range of possible step sizes becomes more limited as β decreases towards 0 (which corresponds to the IS divergence). Conversely, when $d = 2$, we observe that divergences corresponding to β close to 1 (i.e., the KL divergence) allow for more flexibility when it comes to choosing an appropriate step size.

For each setting, we pick the value of the step size that maximizes the SDR on this subset and use it in the following experiment.

4.3 Comparison to other methods

The separation results on the test set are presented in Figure 2. We observe that at high (10 dB) or moderate (0 dB) input SNRs, the proposed algorithm overall outperforms MISI when $d = 2$ and for $\beta \geq 1$. We notably remark a performance peak at around $\beta = 1.25$ depending on the input SNR. This observation is consistent with the findings of [17], where the gradient algorithm using the KL divergence (i.e., $\beta = 1$) in a similar scenario ($d = 2$ and “left” formulation) exhibited good performance.

At low input SNR (-10 dB), the proposed method consistently outperforms the MISI baseline whether $d = 1$ or 2 and for both the “right” and “left” problem formulations. This behavior is somewhat reminiscent of [17]: when the spectrograms are severely degraded (i.e., at low input SNR), the algorithm based on the quadratic loss (here, MISI) is outperformed by algorithms based on more suitable alternative losses. Besides, it is also outperformed by a gradient algorithm based on the same quadratic loss when using a fine-tuned step size. This highlights the potential interest of phase recovery with Bregman divergences in such a scenario.

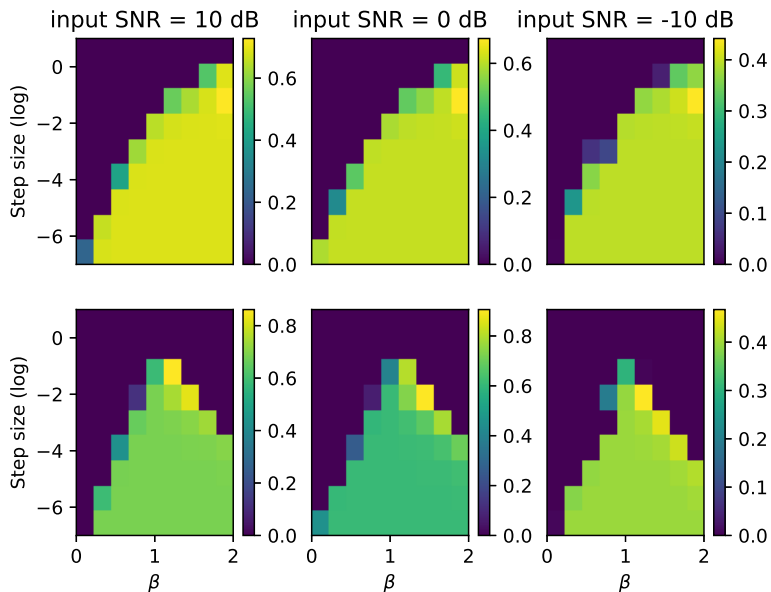


Figure 1: Average SDRi on the validation set obtained with the proposed algorithm at various input SNRs, when $d = 1$ (top) and $d = 2$ (bottom). For better readability, we set the SDRi at 0 when convergence issues occur as visually inspected, or when the SDRi is below 0, as this implies a decreasing performance over iterations, which is not desirable.

Finally, note that the performance of the proposed method strongly depends on the speaker and the kind of noise used in the experiments. Further investigations are needed to identify the optimal β for a given class of signals, which should reduce this sensitivity and improve the above results.

5 Conclusion

In this paper, we have addressed the problem of phase recovery with Bregman divergences for audio source separation. We derived a projected gradient algorithm for optimizing the resulting loss. We experimentally observed that when the spectrograms are highly degraded, some of these Bregman divergences induce better speech enhancement performance than the quadratic loss, upon which the widely-used MISI algorithm builds.

In future work, we will explore other optimization schemes for addressing this problem, such as majorization-minimization or ADMM. We will also leverage these algorithms in a deep unfolding paradigm for end-to-end and time-domain source separation.

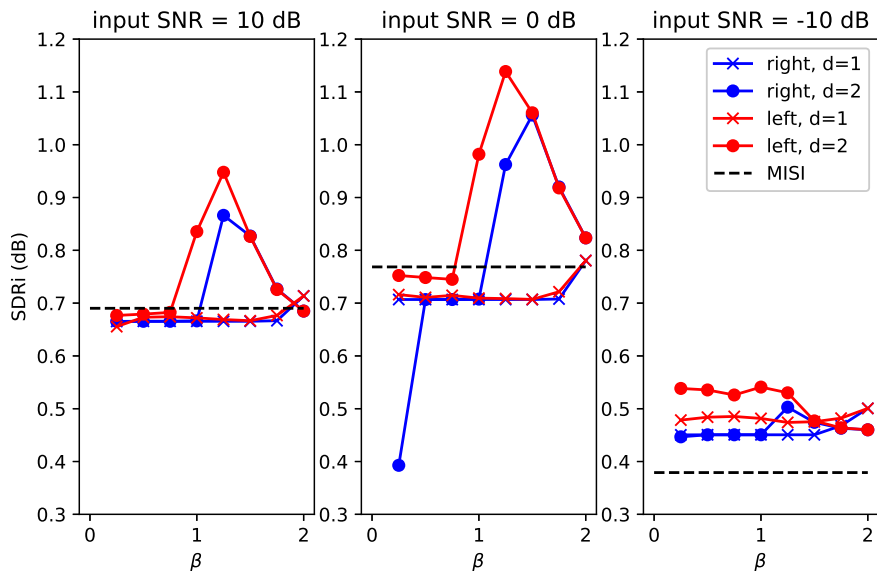


Figure 2: Average SDRi on the test set obtained with MISI and with the proposed algorithm (in different settings) at various input SNRs.

References

- [1] P. Comon and C. Jutten, *Handbook of blind source separation: independent component analysis and applications*, Academic press, 2010.
- [2] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, “The fifth ‘CHiME’ speech separation and recognition challenge: Dataset, task and baselines,” in *Proc. Interspeech*, Sept. 2018.
- [3] E. Cano, D. FitzGerald, A. Liutkus, M. D. Plumbley, and F. Stöter, “Musical source separation: An introduction,” *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 31–40, Jan. 2019.
- [4] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, Oct. 2018.
- [5] Y. Luo and N. Mesgarani, “Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.
- [6] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2020.
- [7] D. Ditter and T. Gerkmann, “A multi-phase gammatone filterbank for speech separation via Tasnet,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2020.
- [8] P. Magron, K. Drossos, S. I. Mimilakis, and T. Virtanen, “Reducing interference with phase recovery in DNN-based monaural singing voice separation,” in *Proc. Interspeech*, Sept. 2018.
- [9] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, “Phase processing for single-channel speech enhancement: History and recent advances,” *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, Mar. 2015.
- [10] Z.-Q. Wang, J. Le Roux, D. Wang, and J. R. Hershey, “End-to-end speech separation with unfolded iterative phase reconstruction,” in *Proc. Interspeech*, Sept. 2018.
- [11] G. Wichern and J. Le Roux, “Phase reconstruction with learned time-frequency representations for single-channel speech separation,” in *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, Sept. 2018.
- [12] S. Wisdom, J. R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R. A. Saurous, “Differentiable consistency constraints for improved deep speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019.
- [13] D. Gunawan and D. Sen, “Iterative phase estimation for the synthesis of separated sources from single-channel mixtures,” *IEEE Signal Processing Letters*, vol. 17, no. 5, pp. 421–424, May 2010.

- [14] R. Gray, A. Buzo, A. Gray, and Y. Matsuyama, “Distortion measures for speech processing,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 367–376, Aug. 1980.
- [15] R. Hennequin, B. David, and R. Badeau, “Beta-divergence as a subclass of Bregman divergence,” *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 83–86, Feb. 2011.
- [16] P. Smaragdis, C. Févotte, G. J. Mysore, N. Mohammadiha, and M. Hoffman, “Static and dynamic source separation using nonnegative factorizations: A unified view,” *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 66–75, May 2014.
- [17] P.-H. Vial, P. Magron, T. Oberlin, and C. Févotte, “Phase retrieval with bregman divergences and application to audio signal recovery,” *submitted to the IEEE Journal of Selected Topics in Signal Processing*, Jan. 2021, <https://arxiv.org/abs/2010.00392>.
- [18] P. L. Combettes and J.-C. Pesquet, “Proximal splitting methods in signal processing,” in *Fixed-point algorithms for inverse problems in science and engineering*, pp. 185–212. Springer, 2011.
- [19] D. Griffin and J. S. Lim, “Signal estimation from modified short-time Fourier transform,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 32, no. 2, pp. 236–243, April 1984.
- [20] T. Qiu, P. Babu, and D. P. Palomar, “PRIME: Phase retrieval via majorization-minimization,” *IEEE Transactions on Signal Processing*, vol. 64, no. 19, pp. 5174–5186, Oct. 2016.
- [21] N. Perraudin, P. Balazs, and P. L. Sondergaard, “A fast Griffin-Lim algorithm,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct. 2013.
- [22] X. Zhu, G. T. Beauregard, and L. L. Wyse, “Real-time signal estimation from modified short-time Fourier transform magnitude spectra,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1645–1653, 2007.
- [23] P. Magron and T. Virtanen, “Online spectrogram inversion for low-latency audio source separation,” *IEEE Signal Processing Letters*, vol. 27, pp. 306–310, 2020.
- [24] J. R. Magnus and H. Neudecker, “Matrix differential calculus with applications to simple, Hadamard, and Kronecker products,” *Journal of Mathematical Psychology*, vol. 29, pp. 474–492, Dec. 1985.
- [25] J. O. Smith, *Spectral audio signal processing*, W3K publishing, 2011.
- [26] C. Valentini-Botinhao, “Noisy speech database for training speech enhancement algorithms and TTS models,” <https://doi.org/10.7488/ds/2117>, 2017.
- [27] J. Thiemann, N. Ito, and E. Vincent, “DEMAND: a collection of multi-channel recordings of acoustic noise in diverse environments,” <https://doi.org/10.5281/zenodo.1227121>, June 2013.
- [28] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, “Open-Unmix - a reference implementation for music source separation,” *Journal of Open Source Software*, 2019.
- [29] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, “Speech enhancement for a noise-robust text-to-speech synthesis system using deep recurrent neural networks,” in *Proc. Interspeech*, Sept. 2016.
- [30] S. Uhlich and Y. Mitsufuji, “Open-unmix for speech enhancement (UMX SE),” <https://doi.org/10.5281/zenodo.3786908>, May 2020.