Fast algorithm for complex-NMF with application to source separation

Andersen Ang[1,3] Valentin Leplat[2] Nicolas Gillis[3]
[1]Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Canada
[2]ICTEAM, Université catholique de Louvain, Louvain-la-Neuve, Belgium
[3]Mathématique et recherche opérationnelle, Université de Mons, Mons, Belgium
ms3ang@uwaterloo.ca, valentin.leplat@uclouvain.be, nicolas.gillis@umons.ac.be

***Abstract*—In this paper we consider a Nonnegative Matrix Factorization (NMF) model on complex numbers, in particular, we propose a group complex-NMF (cNMF) model that subsumes the phase-consistency complex NMF for the audio Blind Source Separation (aBSS). Using Wirtinger calculus, we propose a gradient-based algorithm to solve cNMF. The algorithm is then further accelerated using a heuristic extrapolation scheme. Numerical results show that the accelerated algorithm has significantly faster convergence.**

***Index Terms*—Nonnegative Matrix Factorization, Blind source separation, Phase, Wirtinger calculus, Algorithm, Extrapolation**

I. Introduction

Nonnegative Matrix Factorization (NMF) [1], [2] is the problem of finding two nonnegative matrices, $\mathbf{W}$ and $\mathbf{H}$, from a matrix $\mathbf{X} \in \mathbb{R}^{m\times n}_+$ such that $\mathbf{X} \approxeq \mathbf{WH}$. Factors $\mathbf{W}, \mathbf{H}$ are usually computed by tackling a non-convex optimization problem which are notoriously hard to solve. By extending from $\mathbb{R}^{m\times n}$ to $\mathbb{C}^{m\times n}$, complex NMF (cNMF) introduces more variables, which makes the problems even more difficult to solve. Given $\mathbf{X} \in \mathbb{C}^{m\times n}$ and r, it is defined as follows

$$\min_{\mathbf{W}\in\mathcal{W},\mathbf{H}\in\mathcal{H},\mathbf{\Theta}\in\Omega} D(\mathbf{X}|\mathbf{F}) + \mathcal{R}(\mathbf{F}),\ \mathbf{F} \coloneqq \ell(\mathbf{W},\mathbf{H},\mathbf{e}^{i\mathbf{\Theta}}), \quad \text{(cNMF)}$$

where the sets $\mathcal{W} = \mathbb{R}^{m\times r}_+$, $\mathcal{H} = \mathbb{R}^{r\times n}_+$, $\Omega = [-\pi,\pi]^{m\times n\times G}$ are all convex, the function D measures the distance between $\mathbf{X}$ and $\mathbf{F}$. For simplicity in this paper we focus on the Euclidean norm, using $D(\mathbf{X}|\mathbf{F}) = \frac{1}{2}\|\mathbf{X}-\mathbf{F}\|_F^2$. We refer the reader to [3] for the general case of β-divergences. The term $\mathcal{R}$ is a regularization known as the consistency; which will be discussed in Section II-B.

The estimator $\mathbf{F}$ is computed via the function $\ell : \mathcal{W}\times\mathcal{H}\times\Omega \to \mathbb{C}^{m\times n}$ that produces a matrix to fit $\mathbf{X}$. Three examples of ℓ are:

$$\text{rank-1 1-phase: } \mathbf{F} = \mathbf{WH}\odot\mathbf{e}^{i\mathbf{\Theta}}, \tag{1a}$$

$$\text{rank-1 multi-phase: } \mathbf{F} = \sum_{j=1}^{r}\Big(\mathbf{w}_j\mathbf{h}^j\odot\mathbf{e}^{i\mathbf{\Theta}_j}\Big), \tag{1b}$$

$$\text{grouped multi-phase: } \mathbf{F} = \sum_{j=1}^{G}\Big(\mathbf{W}_{r_j}\mathbf{H}^{r_j}\odot\mathbf{e}^{i\mathbf{\Theta}_j}\Big), \tag{1c}$$

NG acknowledges the support by ERC starting grant No 679515, the Fonds de la Recherche Scientifique and the Fonds Wetenschappelijk Onderzoek under EOS project O005318F-RG47. AA thanks C. Févotte for introducing the works of Le Roux. VL acknowledges the support by ERC Advanced grant No 788368.

where $\mathbf{W},\mathbf{H}$ are real and $\mathbf{e}^{i\mathbf{\Theta}}$ is complex, $\odot$ is the Hadamard product, $i=\sqrt{-1}$ and $\mathbf{e}^{i(\cdot)}$ is the component-wise complex exponential. For fitting the phase angle of $\mathbf{X}$, Model (1a) introduces the angle $\mathbf{\Theta}$. Let $\mathbf{w}_j, \mathbf{h}^j$ denote the jth column of $\mathbf{W}$ and jth row of $\mathbf{H}$, respectively. We have

$$\mathbf{WH}\odot\mathbf{e}^{i\mathbf{\Theta}} = \Big(\sum_{j=1}^{r}\mathbf{w}_j\mathbf{h}^j\Big)\odot\mathbf{e}^{i\mathbf{\Theta}},$$

so (1b) considers the more general case where each rank-1 factor $\mathbf{w}_j\mathbf{h}^j$ has its own phase $\mathbf{e}^{i\mathbf{\Theta}_j}$; this model was introduced in earlier papers on complex NMF [4], [5].

In this paper we introduce (1c), which we refer to as the *grouped multi-phase model* as a natural generalization of (1b); see Section II.

Application cNMF finds applications whose data belongs to the complex domain, a typical example is the audio Blind Source Separation (aBSS) [6]–[8]. It is also related to phase retrieval [9] and finds applications in optics. In this paper we use aBSS as an illustration, which we discuss further in Section II.

Contribution and organization This paper has two contributions. First we consider cNMF with (1c), which generalizes cNMF (1b) studied in [4], [5]; see also [10]. The model (1c) comes naturally from (1b), but it has never been addressed, to the best of our knowledge. Then, the main contribution of this paper comes in the design of a single algorithm framework for solving cNMF that covers the variants in (1). Although $\mathbf{W},\mathbf{H}$ in cNMF are real, in general we can treat $\mathbf{W},\mathbf{H},\mathbf{\Theta}$ as complex variables, and use a systematic approach based on the notion of Wirtinger calculus [11], [12] to design gradient-type algorithm to solve cNMF; see Section III. Furthermore, we adapt the heuristic extrapolation strategy from [13]–[15] to our proposed Wirtinger gradient scheme to solve cNMF, which significantly accelerates its convergence; see Section IV for numerical experiments.

Scope of the paper This paper focuses on algorithmic aspects. We do not benchmark the proposed algorithms with other frameworks on performance w.r.t. aBSS as dedicated data preprocessing, post-processing and parameter tuning are required.

II. cNMF models and aBSS

This section serves as the literature review and background of the paper. We first present how real NMF as in [7], [8] is used to solve single-channel aBSS as shown in Fig.1, then we move to the cNMF previously studied in [4], [5]. Finally

we describe how to generalize these model to the grouped multi-phase model.

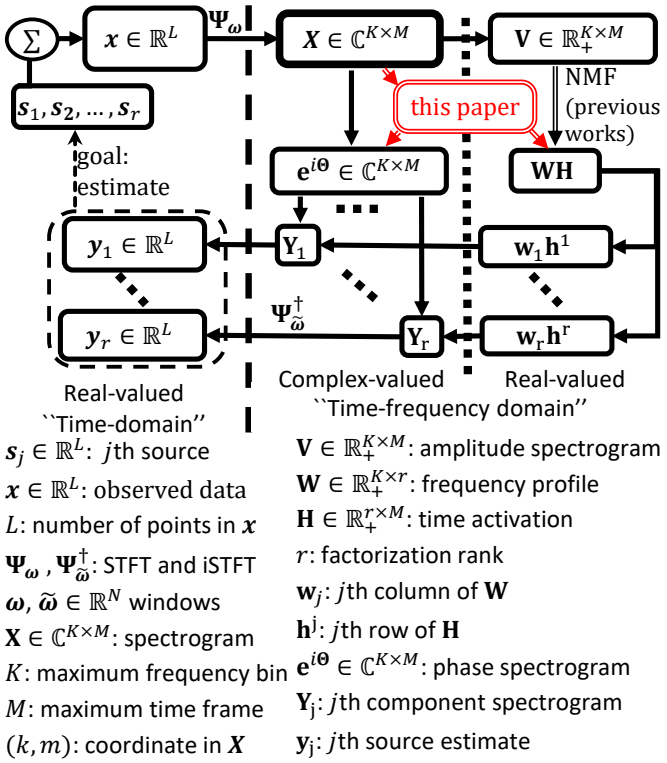

Fig. 1. The processing pipeline as in [7], [8] on using NMF to solve aBSS.

A. Real NMF on single-channel aBSS

Let $[r] := \{1, 2, \cdots, r\}$. Assume r audio sources $\{\mathbf{s}_j \in \mathbb{R}^L\}_{j\in[r]}$ are linearly mixed as $\mathbf{x} = \sum_j \mathbf{s}_j$, aBSS aims to recover $\mathbf{s}_j$ from the observation of their mixture $\mathbf{x} \in \mathbb{R}^L$. We follow the data processing pipeline (Fig. 1) to estimate these sources. First we compute $\mathbf{X} = \boldsymbol{\Psi}_{\boldsymbol{\omega}}(\mathbf{x})$, where $\boldsymbol{\Psi}_{\boldsymbol{\omega}}$ denotes the Short-Time Fourier Transform (STFT) with an analysis window $\boldsymbol{\omega}$, and $\mathbf{X}$ is the spectrogram containing the distribution of "energy" of the signal across time-frequency coordinates. Such information can be extracted by NMF methods through the computation of $\mathbf{W}, \mathbf{H}$ that respectively capture the frequency and temporal patterns [8], [16]. NMF is used here since both the frequency spectrum and time activations are nonnegative.

For aBSS with NMF methods, the input matrix $\mathbf{V}$ typically corresponds to the amplitude spectrogram derived from $\mathbf{X}$ as $V(k, m) = |X(k, m)|$ for all k, m. Then we find $\mathbf{W}, \mathbf{H}$ by solving the (real) NMF problem:

$$[\mathbf{W}, \mathbf{H}] = \min_{\mathbf{W},\mathbf{H}} \frac{1}{2}\|\mathbf{V} - \mathbf{WH}\|_F^2 \text{ s.t. } \mathbf{W} \in \mathcal{W}, \mathbf{H} \in \mathcal{H}. \quad (2)$$

From $\mathbf{WH}$, each rank-1 matrix $\mathbf{w}_j\mathbf{h}^j \in \mathbb{R}_+^{K\times M}$ is converted to a component spectrogram $\mathbf{Y}_j \in \mathbb{C}^{K\times M}$ by multiplying $\mathbf{w}_j\mathbf{h}^j$ element-wise with the phase of the mixture spectrogram $\mathbf{X}$ as follows: $\mathbf{Y}_j = \mathbf{w}_j\mathbf{h}^j \odot \mathbf{e}^{i\boldsymbol{\Theta}}$, where phase spectrogram $e^{i\boldsymbol{\Theta}} \in \mathbb{C}^{K\times M}$ is defined by $\Theta(k, m) = \angle X(k, m)$ with the argument function $\angle$ that returns the angle of the input complex number. Finally, $\boldsymbol{\Psi}_{\tilde{\boldsymbol{\omega}}}^\dagger \mathbf{Y}_j$ is computed to generate the vector $\mathbf{y}_j \in \mathbb{R}^L$, where $\boldsymbol{\Psi}_{\tilde{\boldsymbol{\omega}}}^\dagger$ is the pseudo-inverse of $\boldsymbol{\Psi}$ associated to the synthesis windows $\tilde{\boldsymbol{\omega}}$.

Usual assumptions when using NMF for aBSS Under some conditions we expect the vectors $\mathbf{y}_j$ to accurately estimate the sources $\mathbf{s}_j$. Two of these assumptions are:

- Consistency: each $\mathbf{Y}_j$ is consistent w.r.t. to $\boldsymbol{\Psi}$.
- Rank-1 source: each source is well-approximated by a rank-1 matrix.

Relaxing these assumptions lead to model (1b) and (1c).

B. Complex NMF and consistency

By directly consider factorizing $\mathbf{X}$ instead of $\mathbf{V}$, we arrive at cNMF (1a). We now explain the regularizer term. As Fourier transform is not surjective, this implies that a matrix $\mathbf{P} \in \mathbb{C}^{K\times M}$ does not necessarily corresponds to the STFT of a vector. In that case, we say that $\mathbf{P}$ is not consistent w.r.t. $\boldsymbol{\Psi}_{\boldsymbol{\omega}}$ [5], [10], [17]–[19]. Since the phase is not considered for the computation of $\mathbf{W}, \mathbf{H}$ in (2), it is possible for the aforementioned pipeline to produce inconsistent $\mathbf{Y}_j$, although $\mathbf{X}$ is always consistent. To generate consistent solutions, we can consider the *phase-consistent* [5], [10], [17]–[19] *cNMF*:

$$\min_{\mathbf{W}\in\mathcal{W},\mathbf{H}\in\mathcal{H},\boldsymbol{\Theta}\in\Omega} D(\mathbf{X}|\mathbf{F}) + \mathcal{R}(\mathbf{F}), \quad \mathcal{R}(\mathbf{F}) = \frac{\lambda}{2}\|\mathcal{E}(\mathbf{F})\|_F^2 \quad (3)$$

where λ is a penalty weight (nonnegative scalar), and the term $\mathcal{E}$ is the *phase-inconsistency error* defined as

$$\mathcal{E}(\mathbf{F}) := \mathbf{F} - \boldsymbol{\Psi}_{\boldsymbol{\omega}}\boldsymbol{\Psi}_{\tilde{\boldsymbol{\omega}}}^\dagger\mathbf{F} = \mathbf{BF}, \quad \mathbf{B} := \mathbf{I} - \boldsymbol{\Psi}_{\boldsymbol{\omega}}\boldsymbol{\Psi}_{\tilde{\boldsymbol{\omega}}}^\dagger, \quad (4)$$

we call a matrix $\mathbf{P}$ consistent if $\mathcal{E}(\mathbf{P}) = \mathbf{0}$.

C. Multi-phase and group factorization

We now discuss how (3) can be generalized by cNMF (1b), which is then naturally extended to cNMF (1c).

Multi-phase Note that there is only one $\boldsymbol{\Theta}$ in (3), meaning all the r rank-1 component amplitude spectrograms $\mathbf{w}_j\mathbf{h}^j$ share the same phase. The multi-phase model considers the general case that different sources can have a different phases. In this case, $\boldsymbol{\Theta} \in \mathbb{R}^{m\times n\times r}$ is a 3rd-order tensor, where the jth frontal slice of it, denoted as $\boldsymbol{\Theta}_j$, corresponds to the phase angle of the jth source estimate. This leads to the model cNMF (1b).

Group factorization NMF [7], [8] and cNMF (3) assume each source is well-approximated by a rank-1 matrix in the factorization, and the factorization rank r corresponds to the number of sources. Such an assumption only holds for relatively simple data, and in practice complex real-world audio sources cannot be well-approximated by a rank-1 term. Here we consider using a grouping of rank-1 components to approximate the amplitude part of a source, in this way we arrive at cNMF (1c), where $\mathbf{W}_{r_j}, \mathbf{H}^{r_j}$ are rank-r_j nonnegative matrices with r_j columns or rows, respectively. In this case, we are considering a rank-r NMF with $r = \sum_{i=1}^G r_i$ and G is the total number of groups (number of sources). Immediately, we see a drawback of such model: having more parameters, in particular the parameters $r_1, \cdots, r_G$ which correspond to the model orders. As model order selection is a research topic on its own, in this paper we do not deal with the strategy to tune

these parameters, instead we fix their value for the numerical experiments based on simple prior inspection of the data. For example, a simple heuristic is to use the SVD to numerically check what is the rank of each portion of the data.

Now, we see that real NMF (2) $\subseteq$ cNMF (1a) $=$ (3) $\subseteq$ cNMF (1b) $\subseteq$ cNMF (1c). Thus it is meaningful to study how to effectively solve cNMF, which is the main subject of the next section. Note that fitting a complex matrix is in general harder than fitting a real matrix, this can be easily understood since the equality between two numbers is a stronger assumption on $\mathbb{C}$ than $\mathbb{R}$ as it requires the equality in both modulus and phase. Furthermore, we recall that in [4], [5], sophisticated algorithms based on the MM framework with multiplicative-type update are derived to solve cNMF (1a), which are not suitable here since they cannot directly be used to solve cNMF (1c). Moreover, these algorithms can potentially have slow convergence rates. In fact, multiplicative-type updates are well-known to convergence slower than gradient-based algorithms for NMF when using the Frobenius norm [20]; see also [2], [21] and the references therein. In the next section, we derive a fast gradient-based algorithm to solve cNMF.

III. Algorithms for cNMF models

Here we first review the background on Wirtinger calculus ($\mathbb{W}$-calculus) [11] following similar approach presented in [22, Section 6], and then we present our method to solve cNMF. For details on $\mathbb{W}$-calculus, see [12].

A. $\mathbb{W}$-derivatives on real-valued function of complex variable

In general, functions defined on $\mathbb{C}^n$ are not holomorphic, i.e., they are not complex-differentiable on their domain. A first naive approach would be to consider a related function defined on $\mathbb{R}^{2n}$ such that calculus rules on $\mathbb{R}$ can be used. However, this approach can be rather tedious. $\mathbb{W}$-calculus provides an alternative equivalent formulation, with compact and elegant notation. The core ideas are the $\mathbb{W}$-differential operator and the use of conjugate coordinate $[z, z^*]^\top$ when computing the full gradient. Denote $\partial_z f = \frac{\partial f}{\partial z}$ for a differentiable f and let $z^* = \bar{z}$, the $\mathbb{W}$-derivative w.r.t. $z = x + iy \in \mathbb{C}$, $x, y \in \mathbb{R}$, and its conjugate $z^* \coloneqq x - iy$, respectively, are $\partial_z f =\coloneqq \frac{1}{2}(\partial_x f - i\partial_y f)$, and $\partial_{z^*} f \coloneqq \frac{1}{2}(\partial_x f + i\partial_y f)$. If f is real-valued,

$$(\partial_z f)^* = \partial_{z^*} f. \tag{5}$$

For real-valued multi-variable function $f : \mathbb{C}^n \ni \mathbf{z} = [z_1, z_2 \dots, z_n]^\top \to \mathbf{w} = f(\mathbf{z}) \in \mathbb{R}$, the partial gradient w.r.t. $\mathbf{z}$ and $\mathbf{z}^*$, and the full gradient are:

$$\partial_{\mathbf{z}} f \coloneqq \begin{bmatrix} \partial_{z_1} f \\ \vdots \\ \partial_{z_n} f \end{bmatrix}, \ \partial_{\mathbf{z}^*} f \coloneqq \begin{bmatrix} \partial_{z_1^*} f \\ \vdots \\ \partial_{z_n^*} f \end{bmatrix} \text{ and } \nabla f \coloneqq \begin{bmatrix} \partial_{\mathbf{z}} f \\ \\ \partial_{\mathbf{z}^*} f \end{bmatrix}^*, \tag{6}$$

where the full gradient ∇f consists of the partial gradients $\partial_{\mathbf{z}} f$ and $\partial_{\mathbf{z}^*} f$. As we work on real-valued f, then applying (5) in (6) means that we only need to compute one partial gradient, as the other one is its conjugate. We are now ready to discuss how to solve cNMF.

B. The gradient-update steps

We now discus how to solve cNMF. For simplicity here we focus on cNMF with $\mathbf{F} = \mathbf{WH} \odot \mathbf{e}^{i\boldsymbol{\Theta}}$, that is, (1a). The same approach applies to the other cNMF models. In general, we solve (3) by the (inexact) Block Coordinate Descent (BCD): we alternate on solving each sub-problem on one block of variables while fixing the others at their most recent value. We now discuss how to solve each subproblem.

Subroblem in $\mathbf{W}$ This subproblem has the following form

$$\min_{\mathbf{W} \geq \mathbf{0}} f \coloneqq \frac{1}{2}\|(\mathbf{WH}) \odot \mathbf{D} - \mathbf{X}\|_F^2 + \frac{\lambda}{2}\|\mathbf{B}\big((\mathbf{WH}) \odot \mathbf{D}\big)\|_F^2, \tag{7}$$

where $\mathbf{D} = \mathbf{e}^{i\boldsymbol{\Theta}}$. We solve (7) by iterating

$$\begin{aligned} \mathbf{W}_{k+1} &= \mathbf{Re}\Big\{ \operatorname*{argmin}_{\mathbf{W}} \Big\langle \nabla_{\mathbf{W}} f(\mathbf{W}_k, \mathbf{H}_k, \boldsymbol{\Theta}_k), \mathbf{W} \Big\rangle \\ &\quad + \frac{L_{\mathbf{W}}^k}{2}\|\mathbf{W} - \mathbf{W}_k\|^2 \Big\} + i_+(\mathbf{Re}\{\mathbf{W}\}) \\ &= \Big[\mathbf{Re}\Big\{ \mathbf{W}_k - \frac{1}{L_{\mathbf{W}}^k} \nabla_{\mathbf{W}} f(\mathbf{W}_k) \Big\}\Big]_+ \end{aligned} \tag{8}$$

where $\mathbf{Re}(\,\cdot\,)$ takes the real part, $L_{\mathbf{W}}^k$ is the Lipschitz constant of the gradient at iteration k, i_+ is the indicator function of the nonnegative orthant that encodes the non-negativity constraint, and $[\,\cdot\,]_+ = \max\{\,\cdot\,, \epsilon\}$ with ϵ a small positive value. Using $\mathbb{W}$-derivative, the gradient w.r.t. the Hermitian $\mathbf{W}^{\mathbf{H}}$ is

$$\partial_{\mathbf{W}^{\mathrm{H}}} f = \mathbf{H}\big(\mathbf{D}^{\mathrm{H}} \odot (\mathbf{QF} - \mathbf{X})^\top\big) \coloneqq g(\mathbf{W}^\top), \tag{9}$$

where $\mathbf{Q} \coloneqq \mathbf{I} + \lambda \mathbf{B}^{\mathrm{H}}\mathbf{B}$ and $\mathbf{F}$ is a linear function of $\mathbf{W}$:

$$\mathbf{F} = (\mathbf{WH}) \odot \mathbf{D}. \tag{10}$$

As $\mathbf{W}$ is real, by (5), the gradient of f w.r.t. $\mathbf{W}$ is the transpose of $g(\mathbf{W}^\top)$. For the Lipschitz constant $L_{\mathbf{W}}$ of the gradient (9), let $\Delta_{\mathbf{W}} = \mathbf{W}_1 - \mathbf{W}_2$ and $\Delta g = g(\mathbf{W}_1^\top) - g(\mathbf{W}_2^\top)$, we have

$$\begin{aligned} \|\Delta g\|_2 &\overset{(9)}{=} \left\| \mathbf{H}\Big(\mathbf{D}^{\mathrm{H}} \odot \Big[\mathbf{Q}\Big(\mathbf{F}^\top(\mathbf{W}_1^\top) - \mathbf{F}^\top(\mathbf{W}_2^\top) \Big) \Big] \Big) \right\|_2 \\ &\overset{(10)}{=} \left\| \mathbf{H}\Big(\mathbf{D}^{\mathrm{H}} \odot \Big[\mathbf{Q}\Big(\big(\Delta_{\mathbf{W}}\mathbf{H}\big) \odot \mathbf{D} \Big)^\top \Big] \Big) \right\|_2 \\ &\leq \underbrace{\|\mathbf{H}\|_2^2 \|\mathbf{D}^{\mathrm{H}} \odot \mathbf{D}^\top\|_2 \|\mathbf{Q}\|_2}_{L_{\mathbf{W}}} \|\Delta_{\mathbf{W}}\|_2. \end{aligned} \tag{11}$$

Subproblem in $\mathbf{H}$ By taking transpose, the subproblem on $\mathbf{H}$ is symmetric to (7), we use the same approach to solve it.

Suproblem in $\boldsymbol{\Theta}$ Based on (7), the subproblem in $\boldsymbol{\Theta}$ is

$$\min_{\mathbf{D} \in \mathcal{D}} f(\mathbf{D}) \coloneqq \frac{1}{2}\|\mathbf{A} \odot \mathbf{D} - \mathbf{X}\|_F^2 + \frac{\lambda}{2}\|\mathbf{B}(\mathbf{A} \odot \mathbf{D})\|_F^2, \tag{12}$$

where $\mathbf{A} = \mathbf{WH}$ and $\mathbf{D} = \mathbf{e}^{i\boldsymbol{\Theta}}$. Note that we use the change of variable $\mathbf{D}$ to replace $\boldsymbol{\Theta}$ as the variable, and note that since complex exponential has unit norm, this requires an additional constraint $\big|[\mathbf{D}]_{ij}\big| = 1$, so that $\mathcal{D}$ in (12) is the set of complex matrices whose elements have a magnitude equal to one. Note that the set $\mathcal{D}$ is nonconvex, which affects the convergence analysis; see the discussion in Section III-C.

To solve (12) and update $\mathbf{D}$, we follow the same approach as for (8), and use

$$\mathbf{D}_{k+1} = P_{\mathcal{D}}\Big\{\mathbf{D}_k - \frac{1}{L_{\mathbf{D}}^k}\nabla_{\mathbf{D}} f(\mathbf{D}_k)\Big\}, \tag{13}$$

where $P_{\mathcal{D}}$ is the projection onto the set $\mathcal{D}$. Take $\mathbf{D}$ as $\mathbf{W}$ in (7) with $\mathbf{H}$ set to $\mathbf{I}$, the the same update (9) can be used with $L_{\mathbf{D}} = \|\mathbf{A}\odot\mathbf{A}\|_2\|\mathbf{Q}\|_2$. Lastly, we get $\mathbf{\Theta}_k$ from $\mathbf{D}_k$ by $\mathbf{\Theta} = \angle\mathbf{D}$.

C. The overall algorithm and convergence

Algorithm 1 shows the simplified pseudo-code of the overall algorithm. At first glance it looks tempting to apply some well-established convergence analyses for, say proximal gradient method and block coordinate descent to derive the theoretical convergence of the algorithm. However, we emphasize that the algorithm is a heuristic and we currently do not have a rigorous proof of convergence. It is important to note that cNMF involves complex variables, while the aforementioned convergence analyses are all built upon the assumption that the objective function sits in $\mathbb{R}$, hence it is unclear how to apply convergence analysis from the real case on the complex case. Furthermore, if it turns out the "real analysis" can be applied on such a complex problem, the convergence analysis is still nontrivial as the update step of $\mathbf{D}$ involves a projection onto a nonconvex set, leading to theoretical complications from the projection $P_{\mathcal{D}}$ in the update (13).

Although we do not have theoretical convergence, we have observed empirically that the algorithm decreases the objective function; see Section IV and in particular Figure 3.

Algorithm 1: Alternating Projected Wirtinger gradient

Result: $\mathbf{W}, \mathbf{H}, \mathbf{\Theta}$ that solves cNMF (1a)
Initialize $\mathbf{W}_0, \mathbf{H}_0, \mathbf{\Theta}_0, \mathbf{D}_0 = \mathbf{e}^{i\mathbf{\Theta}_0}, \mathbf{Q} = \mathbf{I} + \lambda\mathbf{B}^{\mathrm{H}}\mathbf{B}$;
for $k = 1, 2, \ldots$ **do**
  Get $\mathbf{W}_{k+1}$ using (8) and (11);
  Get $\mathbf{H}_{k+1}$, $\mathbf{D}_{k+1}$ similarly as $\mathbf{W}$;
  Get $\mathbf{\Theta}_{k+1}$ from $\mathbf{D}_{k+1}$;
end

D. Acceleration by heuristic extrapolation

Algorithm 1 can be accelerated using extrapolation, and we adopt the framework of Heuristic Extrapolation and Restart (HER) [13]–[15], which has been shown to be very effective on accelerating the convergence of NMF-type algorithms with the Frobenius norm. In a nutshell, HER is a numerical extrapolation scheme based on the extrapolation on the sequence $\{\mathbf{W}_k, \mathbf{H}_k\}_{k=1,2,\ldots}$ using an auxiliary sequence $\{\hat{\mathbf{W}}_k, \hat{\mathbf{H}}_k\}_{k=1,2,\ldots}$ as follows:

$$\begin{aligned}
\mathbf{W}_{k+1} &= \texttt{Update}(\hat{\mathbf{W}}_k, \hat{\mathbf{H}}_k),\\
\hat{\mathbf{W}}_{k+1} &= \big[\mathbf{W}_{k+1} + \beta_k(\mathbf{W}_{k+1} - \mathbf{W}_k)\big]_+,\\
\mathbf{H}_{k+1} &= \texttt{Update}(\hat{\mathbf{W}}_{k+1}, \hat{\mathbf{H}}_k),\\
\hat{\mathbf{H}}_{k+1} &= \big[\mathbf{H}_{k+1} + \beta_k(\mathbf{H}_{k+1} - \mathbf{H}_k)\big]_+,
\end{aligned}$$

where `Update` can be the gradient update as in (8), and β_k is the extrapolation parameter automatically tuned based on the HER scheme. The acceleration effect of HER comes from the combination of extrapolation, cheap restart (safe guard mechanism) and a numerical scheme on updating the extrapolation weight β_k; see [14] for more details. Lastly, we emphasize that currently there is no theoretical convergence guarantee for the HER framework, but HER empirically works well in practice; see Section IV.

Remark on HER In principle we can also extrapolate the variable $\mathbf{D}$ in the same way as $\mathbf{W}$ and $\mathbf{H}$. However we do not perform extrapolation on the variable $\mathbf{D}$. The reason is that we empirically observed that not extrapolating $\mathbf{D}$ performs better, although extrapolating $\mathbf{D}$ is still much faster than the unextrapolated original gradient-descent algorithm.

Note that currently the theoretical understanding of the HER mechanism is still very limited. Let us try to give a partial explanation of the ineffectiveness of extrapolating $\mathbf{D}$ in the HER framework on cNMF. Since $\mathbf{D}$ is to be projected element-wise such that $\big|[\mathbf{D}]_{ij}\big| = 1$, that is, all the elements of $\mathbf{D}$ sits on the unit circle on the complex plane, then the effect of extrapolating $\mathbf{D}$ ultimately is just a rotation on the elements of $\mathbf{D}$, and we hypothesize that the HER framework is not a good set up for performing this kind of rotational extrapolation, since it was original proposed for speed up NMF computation in the Frobenius norm in the Euclidean geometry on the rectangular coordinate system. Understanding how to extrapolate $\mathbf{D}$ effectively is an interesting future research topic.

IV. Experiment

We present numerical results to confirm the effectiveness of the proposed algorithm, and to show case the effectiveness of the acceleration. The code is available at `angms.science`. Before we go to the details of the experiment, we remark that:

- As mentioned in the beginning of the paper, the task of aBSS itself contains highly involved data processing pipeline such as parameter tuning, data pre-processing, filtering and post-processing, which is not the focus of this paper, so we do not focus on these issues and demonstrate that the proposed algorithm has a fast convergence in practice. We set $\lambda = \frac{D(\mathbf{X}|\mathbf{F}_0)}{\mathcal{R}(\mathbf{F}_0)}$ in all the experiments, where $\mathbf{F}_0 = \ell(\mathbf{W}_0, \mathbf{H}_0, \mathbf{e}^{i\mathbf{\Theta}_0})$ and $\mathbf{W}_0, \mathbf{H}_0, \mathbf{\Theta}_0$ are initialization of the variables $\mathbf{W}, \mathbf{H}, \mathbf{\Theta}$. This makes the two terms in the objective function well balanced.
- We do not benchmark with the algorithms proposed in [4], [5] because the codes are unavailable. See also the discussion in the last paragraph in Section II-C.

We run Algorithm 1 (with and without HER, with the same random initialization) on two data sets (see Fig. 2), which are both real-world recording with some ambient noise: "Mary", $\mathbf{X} \in \mathbb{C}^{256\times 714}$, a piano music phrase "Mary had a little lamb"; and "Voice", $\mathbf{X} \in \mathbb{C}^{256\times 162}$, a speech of the letters "NMF". We run cNMF (1a) on Mary with $r \in \{3, 6\}$ and on Voice with $r = 75$. We run group cNMF (1c) on Voice with $G = 3$ and two sets of r_j: $r_1 = r_2 = r_3 = 15$ and $r_1 = r_2 = r_3 = 25$,

(recall that $r = \sum_{j=1}^{G} r_j$). All the experiments are repeated 20 times with different random initialization of $(\mathbf{W}_0, \mathbf{H}_0, \mathbf{\Theta}_0)$.

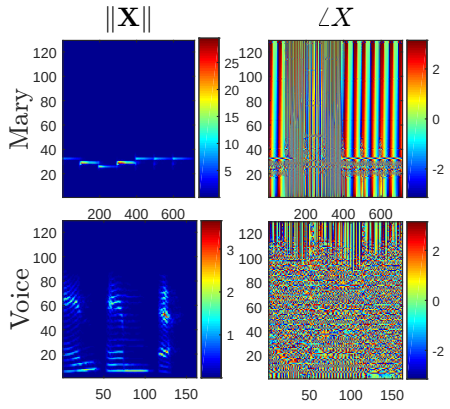

Fig. 2. The spectrograms of the two datasets Mary and Voice.

Fig. 3 and Fig. 4 show the convergence results. First we see that the HER framework significantly accelerates the convergence. Second, comparing the result on cNMF (1a) on Voice with $r = 75$ to group cNMF (1c) on Voice with $G = 3, r_1 = r_2 = r_3 = 25$ so $r = \sum_{j=1}^{G} r_j = 75$, the group models have a lower objective function value. This is expected as group model uses more phase variables to fit the data.

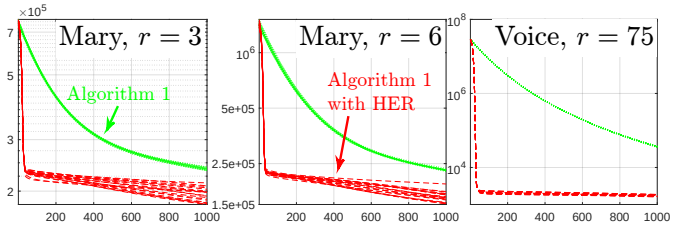

Fig. 3. Convergence of the objective function values $\mathcal{D} + \mathcal{R}$ on cNMF tests. x-axes are iteration. We do not show the time plots as the cost per iteration of both algorithm is almost identical (HER adds a negligible cost [13]).

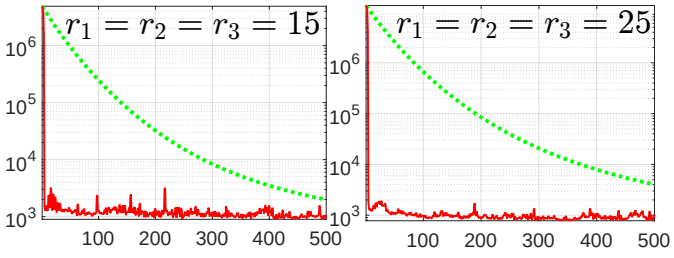

Fig. 4. Convergence curves on group cNMF tests. Here the curves are the mean over 20 experiments.

V. Conclusion

In this paper, we introduced the group cNMF model that subsumes the existing cNMF and real NMF models. Using Wirtinger calculus, we derived a general gradient-based algorithm to solve cNMF. By using the heuristic extrapolation with restart, we showed that, on a few preliminary numerical tests, the accelerated algorithm has a much faster convergence.

Future works include studying the convergence of the algorithm, performing tests with respect to the task of aBSS, and consider replacing the Frobenius norm in the data fitting term by β-divergences which is more appropriate for audio data sets, and also study identifiability issues as in [8].

References

[1] S. A. Vavasis, "On the complexity of nonnegative matrix factorization," *SIAM Journal on Optimization*, vol. 20, no. 3, pp. 1364–1377, 2010.

[2] N. Gillis, *Nonnegative Matrix Factorization*. SIAM, Philadelphia, 2020.

[3] P. Magron and T. Virtanen, "Towards complex nonnegative matrix factorization with the beta-divergence," in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 156–160.

[4] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, "Complex nmf: A new sparse representation for acoustic signals," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 3437–3440.

[5] J. Le Roux, H. Kameoka, E. Vincent, N. Ono, K. Kashino, and S. Sagayama, "Complex nmf under spectrogram consistency constraints," in *Acoustical Society of Japan Autumn Meeting*, 2009.

[6] P. Magron, R. Badeau, and B. David, "Phase recovery in nmf for audio source separation: an insightful benchmark," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 81–85.

[7] V. Leplat, N. Gillis, X. Siebert, and A. M. S. Ang, "Séparation aveugle de sources sonores par factorization en matrices positives avec pénalité sur le volume du dictionnaire," in *XXVIIème Colloque francophone de traitement du signal et des images*. GRETSI 2019, 2019.

[8] V. Leplat, N. Gillis, and A. M. S. Ang, "Blind audio source separation with minimum-volume beta-divergence nmf," *IEEE Transactions on Signal Processing*, 2020.

[9] D. Lahat and C. Févotte, "Positive semidefinite matrix factorization: A link to phase retrieval and a block gradient algorithm," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 5705–5709.

[10] P. Magron, R. Badeau, and B. David, "Complex nmf under phase constraints based on signal modeling: application to audio source separation," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 46–50.

[11] W. Wirtinger, "Zur formalen theorie der funktionen von mehr komplexen veränderlichen," *Mathematische Annalen*, vol. 97, no. 1, pp. 357–375, 1927.

[12] P. Bouboulis, "Wirtinger's calculus in general hilbert spaces," *arXiv preprint arXiv:1005.5170*, 2010.

[13] A. M. S. Ang and N. Gillis, "Accelerating nonnegative matrix factorization algorithms using extrapolation," *Neural computation*, vol. 31, no. 2, pp. 417–439, 2019.

[14] A. M. S. Ang, J. E. Cohen, N. Gillis, and L. T. K. Hien, "Extrapolated alternating algorithms for approximate canonical polyadic decomposition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3147–3151.

[15] ——, "Accelerating block coordinate descent for nonnegative tensor factorization," *Numerical Linear Algebra with Applications*, 2021.

[16] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (IEEE Cat. No. 03TH8684)*. IEEE, 2003, pp. 177–180.

[17] J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, "Fast signal reconstruction from magnitude stft spectrogram based on spectrogram consistency," in *International Conference on Digital Audio Effects*, 2010.

[18] J. Bronson and P. Depalle, "Phase constrained complex nmf: Separating overlapping partials in mixtures of harmonic musical sources," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 7475–7479.

[19] D. Kitamura and K. Yatabe, "Consistent independent low-rank matrix analysis for determined blind source separation," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 1, pp. 1–35, 2020.

[20] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[21] N. Gillis and F. Glineur, "Accelerated multiplicative updates and hierarchical als algorithms for nonnegative matrix factorization," *Neural computation*, vol. 24, no. 4, pp. 1085–1105, 2012.

[22] E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.