MAJORIZATION-MINIMIZATION ALGORITHMS FOR CONVOLUTIVE NMF WITH THE BETA-DIVERGENCE

Dylan Fagot, Herwig Wendt, Cédric Févotte[*]

IRIT, Université de Toulouse, CNRS
Toulouse, France

Paris Smaragdis[†]

University of Illinois at Urbana-Champaign
Adobe Research

ABSTRACT

Nonnegative matrix factorization (NMF) has become a method of choice for spectrogram decomposition. However, its inability to capture dependencies across columns of the input motivated the introduction of a variant, convolutive NMF. While algorithms for solving the convolutive NMF problem were previously proposed, they rely on the use of a heuristic that does not insure the convergence of the algorithm (in particular in terms of objective function values). The goal of this work is to propose rigorous update rules, based on a majorization-minimization (MM) approach, for convolutive NMF with the β-divergence (a standard family of measures of fit). Specifically, we derive and study two variants of a convolutive NMF algorithm that are guaranteed to decrease the objective function value at each iteration. The complexity of the algorithms is studied, and the performance in terms of execution time and objective function are evaluated and compared in several numerical experiments using real-world audio data. Experiments show that the proposed MM algorithms consistently provide lower values of the objective function than the heuristic, at similar computational cost.

Index Terms— Nonnegative matrix factorization (NMF), majorization-minimization (MM)

1. INTRODUCTION

Nonnegative matrix factorization (NMF) consists of decomposing nonnegative data $\mathbf{V} \in \mathbb{R}_+^{M \times N}$, such as a spectrogram, into

$$\mathbf{V} \approx \mathbf{WH} \tag{1}$$

where $\mathbf{W} \in \mathbb{R}_+^{M \times K}$ and $\mathbf{H} \in \mathbb{R}_+^{K \times N}$ are two nonnegative matrices referred to as dictionary and activation matrix, respectively. K is usually chosen such that the decomposition is low-rank ($K < \min(M, N)$). NMF is known to produce a factorization that gives a part-based representation of $\mathbf{V}$. This method, popularized by Lee and Seung [1], has led to state-of-the-art results in audio source separation [2, 3, 4] and music transcription [5, 6]. In the context of audio processing, NMF is typically applied on a spectrogram, with each column corresponding to one time frame of data. It can lead to a meaningful decomposition where the dictionary tends to capture the vertical structure (spectral patterns) while the activation matrix encodes how these are mixed.

In audio data, it is common that spectral patterns (e.g., a note) extend over several time frames. However, by construction, $\mathbf{W}$ in the factorization (1) cannot model such temporal dependencies among the columns of $\mathbf{V}$. This limitation of NMF (1) (which will be termed *traditional NMF* hereafter to avoid ambiguities) has motivated the introduction of a variant, the so-named *convolutive NMF*, which is able to capture these temporal dependencies efficiently and encode them in a single pattern. Convolutive NMF computes

$$\mathbf{V} \approx \hat{\mathbf{V}} \triangleq \sum_{t=0}^{T-1} \mathbf{W}(t) \overset{t\rightarrow}{\mathbf{H}} \tag{2}$$

where $\{\mathbf{W}(t)\}_t \in \mathbb{R}_+^{M \times K}$ are nonnegative matrices referred to as dictionaries. The notation $\overset{t\rightarrow}{\cdot}$ denotes the operator which shifts the entries of a matrix by t columns to the right and replaces the ones shifted in from outside the matrix with zeros. In other words, each column of $\hat{\mathbf{V}}$ is given by a convolution $\hat{\mathbf{v}}_n = \sum_t \mathbf{W}(t)\mathbf{h}_{n-t}$ where $\mathbf{h}_n$ denotes the n-th column of $\mathbf{H}$. The columns $\mathbf{w}_k$ of the dictionary $\mathbf{W}$ in traditional NMF now become patches

$$\mathbf{P}_k = [\mathbf{w}_k(0), \ldots, \mathbf{w}_k(T-1)] \in \mathbb{R}_+^{M \times T} \tag{3}$$

which capture patterns of length T.

Convolutive NMF was originally introduced in [7]. In that work, the author also proposed an algorithm for finding an approximation $\hat{\mathbf{V}}$ as in (2), by minimization of an objective function of the form $D(\mathbf{V}|\hat{\mathbf{V}})$, where $D(\cdot|\cdot)$ is a measure of fit, cf. (4) below. However, the update rule for $\mathbf{H}$ in [7] is based on a heuristic, see Section 3.2 for details. This constitutes an important limitation because it leads to an algorithm that does not guarantee the decrease of the objective function value. Moreover, the update rules were derived only for a variant of the Kullback-Leibler divergence. Other authors [8] derived update rules for convolutive NMF based on the Euclidean distance criterion but use the same kind of heuristic to update $\mathbf{H}$.

Goals and contributions. The goal of this paper is to provide rigorous algorithms for solving the convolutive NMF problem (briefly recalled in Section 2) that guarantee a decrease of the objective function values between two iterates. Our approach is based on the majorization-minimization (MM) technique [9], which is widely used in the context of NMF and leads to inexpensive update rules. Specifically, we derive two new update rules for $\mathbf{H}$ that ensure the convergence of the objective function value (cf. Section 3). The new algorithms are found to have identical leading-order complexity $\mathcal{O}(MKNT)$ as the previous, heuristic formulation, yet to lead to significant practical benefits in terms of obtained objective function values (cf. Section 4). MATLAB codes for the proposed novel algorithms are made available online.

2. CONVOLUTIVE NMF

The goal of convolutive NMF is to minimize some measure of fit between the data and the approximate factorization (2). Finding latent

[*]Supported by ERC grant #681839 (European Union's Horizon 2020 research and innovation program, project FACTORY).

[†]Supported by NSF grant #1453104.

Algorithm 1: Convolutive NMF

Input : $\mathbf{V}$, K, β
Output: $\{\mathbf{W}(t)\}_t, \mathbf{H}$

Initialize $\mathbf{H}$ and $\mathbf{W}(t),\ t = 1, \dots, T$
for $i = 1 : N_{iter}$ **do**
  Update $\mathbf{H}$
  Update $\mathbf{W}(t),\ t = 1, \dots, T$
  Normalize $\mathbf{H}$ and $\mathbf{W}(t),\ t = 1, \dots, T$
end

factors $\{\mathbf{W}(t)\}_t$ and $\mathbf{H}$ satisfying (2) is achieved by solving

$$\min_{\{\mathbf{W}(t)\}_t, \mathbf{H}} D\left(\mathbf{V} \middle| \sum_{t=0}^{T-1} \mathbf{W}(t) \overset{t\rightarrow}{\mathbf{H}}\right) \text{ s.t. } \forall t\ \mathbf{W}(t) \geq 0, \mathbf{H} \geq 0 \quad (4)$$

where $D(\cdot|\cdot)$ is a measure of fit. This paper will assume that D belongs to the family of the β-divergences [10, 11], which is popular in NMF and encompasses the squared Euclidean distance, the generalized Kullback-Leibler (KL) divergence and the Itakura-Saito (IS) divergence for values of β of 2, 1 and 0, respectively.

Note that, as for traditional NMF, convolutive NMF suffers from a scale indeterminacy as for any solution $(\{\mathbf{W}(t)\}_t, \mathbf{H})$ of the problem in (4), $(\{\mathbf{W}(t)\mathbf{\Lambda}^{-1}\}_t, \mathbf{\Lambda}\mathbf{H})$ is also a solution, where $\mathbf{\Lambda}$ is a $K \times K$ diagonal matrix, with diagonal elements $\lambda_k > 0$. Similarly to traditional NMF, we can define a renormalization matrix as $\mathbf{\Lambda} = \text{diag}\left(||\mathbf{P}_1||_1, ..., ||\mathbf{P}_K||_1\right)$. The normalization $\forall t\ \mathbf{W}(t) \leftarrow \mathbf{W}(t)\mathbf{\Lambda}^{-1}$, $\mathbf{H} \leftarrow \mathbf{\Lambda}\mathbf{H}$, results in patches which have their ℓ_1-norm equal to 1.

Likewise [7], we propose to resort to a block-coordinate descent algorithm that alternates between updates for the latent factors $\{\mathbf{W}(t)\}_t$ and $\mathbf{H}$ to solve (2). The global algorithm to solve the convolutive NMF problem can be found in Algorithm 1. In the following, we study in detail the update steps for $\{\mathbf{W}(t)\}_t$ and $\mathbf{H}$.

3. ALGORITHMS FOR UPDATING $\{\mathbf{W}(t)\}_t$ AND $\mathbf{H}$

3.1. Update rule for $\{\mathbf{W}(t)\}_t$

Unlike traditional NMF, convolutive NMF involves T dictionaries denoted $\{\mathbf{W}(t)\}_t$. Yet, minimizing $D(\mathbf{V}|\hat{\mathbf{V}})$ for a given dictionary $\mathbf{W}(t)$ with $\mathbf{H}$ and $\{\mathbf{W}(\tau)\}_{\tau \neq t}$ fixed leads to minimizing

$$D\left(\mathbf{V} \middle| \mathbf{W}(t) \overset{t\rightarrow}{\mathbf{H}} + \sum_{\tau \neq t} \mathbf{W}(\tau) \overset{\tau\rightarrow}{\mathbf{H}}\right) \quad t = 0, \dots, T-1, \quad (5)$$

which, for each $t = 1, \dots, T$, can be seen as a NMF subproblem with a residual term, i.e., the minimization of a function of the form $D(\mathbf{V}|\mathbf{W}\mathbf{H} + \mathbf{R})$ with $\mathbf{H}$ and $\mathbf{R}$ fixed. Using the results of [12], the MM update for $\mathbf{W}(t)$ is therefore given by

$$\mathbf{W}(t) \leftarrow \mathbf{W}(t) \circ \left(\frac{\left(\mathbf{V} \circ \hat{\mathbf{V}}^{\circ(\beta-2)}\right) \overset{t\rightarrow}{\mathbf{H}}^T}{\hat{\mathbf{V}}^{\circ(\beta-1)} \overset{t\rightarrow}{\mathbf{H}}^T} \right)^{\circ\gamma(\beta)}, \quad (6)$$

where $\circ$ denotes the entry-wise product/exponentiation, $\gamma(\beta) = (2-\beta)^{-1}\mathbb{1}_{\{\beta<1\}} + \mathbb{1}_{\{1\leq\beta\leq2\}} + (\beta-1)^{-1}\mathbb{1}_{\{\beta\geq2\}}$ and $\mathbb{1}_{\mathcal{I}}$ is the indicator function of set $\mathcal{I}$. In the remaining of the paper, the dictionaries will always be updated using (6).

3.2. Heuristic update for $\mathbf{H}$

Unlike the dictionaries $\{\mathbf{W}(t)\}_t$, the NMF subproblems (5) are tied by a unique (shifted) activation matrix $\mathbf{H}$. The update rule for $\mathbf{H}$ proposed in [7] proposes to bypass this difficulty by introducing T uncoupled surrogate activation matrices $\bar{\mathbf{H}}(t)$ that replace $\overset{t\rightarrow}{\mathbf{H}}$ in (5). The set of surrogate activation matrices are updated independently and then averaged to form the new update of $\mathbf{H}$. The update for the t-th surrogate activation matrix $\bar{\mathbf{H}}(t)$ is given by the traditional NMF (with residual) update rule

$$\bar{\mathbf{H}}(t) \leftarrow \mathbf{H} \circ \left(\frac{\mathbf{W}^T(t) \left[\overset{\leftarrow t}{\mathbf{V}} \circ \left(\overset{\leftarrow t}{\hat{\mathbf{V}}} \right)^{\circ(\beta-2)} \right]}{\mathbf{W}^T(t) \left(\overset{\leftarrow t}{\hat{\mathbf{V}}} \right)^{\circ(\beta-1)}} \right)^{\circ\gamma(\beta)}, \quad (7)$$

which we state here for the β-divergence [12]. The average then writes

$$\mathbf{H} \leftarrow \langle \bar{\mathbf{H}}(t) \rangle_t. \quad (8)$$

The artificial de-coupling using surrogate matrices $\bar{\mathbf{H}}(t)$ implies that there is no guarantee for a monotonous decrease of the objective function. We can thus expect that the objective function may not decrease at each step, and experiments confirm that this happens in practice (see Section 4). Algorithm 2 sums up the heuristic procedure to update $\mathbf{H}$.

3.3. Sequential MM update for $\mathbf{H}$

We now propose an update rule following a formal MM approach that respects the convolutive structure of the problem. This is achieved by updating the columns of $\mathbf{H}$ sequentially (forward pass). The objective function is

$$D(\mathbf{V}|\hat{\mathbf{V}}) = \sum_{n=1}^{N} D\left(\mathbf{v}_n \middle| \sum_{t=0}^{\rho(n)} \mathbf{W}(t)\mathbf{h}_{n-t}\right) \quad (9)$$

with $\rho(n) = (n-1)\mathbb{1}_{\{n<T\}} + (T-1)\mathbb{1}_{\{n\geq T\}}$. Minimizing the divergence in (9) with respect to a particular column $\mathbf{h}_n$ of $\mathbf{H}$ amounts to minimizing

$$\begin{aligned} C_n(\mathbf{h}_n) &= \sum_{n' \in \mathcal{I}(n)} D\left(\mathbf{v}_{n'} \middle| \sum_{t=0}^{\rho(n')} \mathbf{W}(t)\mathbf{h}_{n'-t}\right) \\ &= \sum_{n' \in \mathcal{I}(n)} D\left(\mathbf{v}_{n'} \middle| \mathbf{W}(n'-n)\mathbf{h}_n + \mathbf{b}(n', n)\right) \quad (10) \end{aligned}$$

with $\mathbf{b}(n', n) = \sum_{t \neq (n'-n)} \mathbf{W}(t)\mathbf{h}_{n'-t}$ and $\mathcal{I}(n) = [n, \min\{n + T - 1, N\}]$. The vector $\mathbf{b}(n', n)$ does not depend on $\mathbf{h}_n$ but only on (the current value of) neighboring columns. Finding an upper bound to $C_n(\mathbf{h}_n)$ can be done by bounding the individual summands in (10) and summing the bounds. The summands are given by

$$\begin{aligned} &D(\mathbf{v}_{n'}|\mathbf{W}(n'-n)\mathbf{h}_n + \mathbf{b}(n', n)) = \\ &\qquad \sum_m D\left(v_{mn'} \middle| \sum_k w_{mk}(n'-n)h_{kn} + b_m(n', n)\right). \end{aligned} \quad (11)$$

The summands in (11) can themselves be decomposed as the sum of convex and concave parts which can be bounded with Jensen and

Algorithm 2: Three different algorithms to update $\mathbf{H}$

Input : $\mathbf{V}$, $\hat{\mathbf{V}}$, $\{\mathbf{W}(t)\}_t$, $\mathbf{H}$, β
Output: $\mathbf{H}$

Variant 1 - Heuristic:
Compute each $\bar{\mathbf{H}}(t)$ as in (7)
Update $\mathbf{H}$ as in (8)
Update $\hat{\mathbf{V}}$

Variant 2 - MM1:
for $n = 1 : N$ **do**
  Update $\mathbf{h}_n$ as in (12)
  Update the Card($\mathcal{I}(n)$) columns of $\hat{\mathbf{V}}$ altered by $\mathbf{h}_n$
end

Variant 3 - MM2:
for $n = 1 : N$ **do**
  Update $\mathbf{h}_n$ as in (12)
end
Update $\hat{\mathbf{V}}$

tangent inequalities, following [10, 12]. The overall resulting bound can be minimized analytically, leading to

$$\mathbf{h}_n \leftarrow \mathbf{h}_n \circ \left(\frac{\sum_{n' \in \mathcal{I}(n)} \mathbf{W}^T(n'-n)(\mathbf{v}_{n'} \circ \hat{\mathbf{v}}_{n'}^{\circ(\beta-2)})}{\sum_{n' \in \mathcal{I}(n)} \mathbf{W}^T(n'-n)\hat{\mathbf{v}}_{n'}^{\circ(\beta-1)}} \right)^{\circ\gamma(\beta)}. \quad (12)$$

It is easy to see that the update rule is consistent with traditional NMF. Indeed, when setting $T = 1$, (12) boils down to the traditional NMF multiplicative MM update [10].

The block-coordinate descent architecture dictates that we update $\hat{\mathbf{V}}$ after each update of a column of $\mathbf{H}$. It is unnecessary to update the whole $\hat{\mathbf{V}}$ after updating $\mathbf{h}_n$ since only the columns n to $\min\{n + T - 1, N\}$ of $\hat{\mathbf{V}}$ are altered by the update of $\mathbf{h}_n$. This method for updating $\mathbf{H}$, summarized in Algorithm 2, will be called MM1 thereafter.

3.4. Sequential formulation of the global MM update for H

It can be shown that the convolutive NMF problem can be solved by casting it as a traditional NMF problem by unfolding the convolution in a higher dimensional space. Indeed, introducing vec$(\cdot)$ the vectorization operator and $\otimes$ the Kronecker product, (2) can be written as $\text{vec}(\mathbf{V}) \approx \mathcal{W}\,\text{vec}(\mathbf{H})$ using the property $\text{vec}(\mathbf{AB}) = (\mathbf{I} \otimes \mathbf{A})\text{vec}(\mathbf{B})$ where

$$\mathcal{W} = \sum_{t=0}^{T-1} \overset{t\swarrow}{\mathbf{I}}_N \otimes \mathbf{W}(t) \quad (13)$$

and $\overset{t\swarrow}{\mathbf{I}}_N$ denotes the $N \times N$ matrix of zeros with a t-th subdiagonal of ones. $\mathcal{W}$ is a $MN \times KN$ block-band matrix resulting from the sum of all Kronecker products. This matrix is nonnegative and can thus play the role of the dictionary in a traditional NMF problem. Using this formulation, the activation $\mathbf{H}$ can be updated with standard MM [10], leading to

$$\text{vec}(\mathbf{H}) \leftarrow \text{vec}(\mathbf{H}) \circ \left(\frac{\mathcal{W}^T \text{vec}\left(\mathbf{V} \circ \hat{\mathbf{V}}^{\circ(\beta-2)}\right)}{\mathcal{W}^T \text{vec}(\hat{\mathbf{V}}^{\circ(\beta-1)})} \right)^{\circ\gamma(\beta)}. \quad (14)$$

β	Algorithm	$MKNT$	MNT	KNT	MKT
0	Heuristic	12	6	2	2
	MM1	12	5	0	3
	MM2	12	5	0	3
1	Heuristic	8	4	3	3
	MM1	9	5	0	2
	MM2	9	5	0	2
2	Heuristic	12	0	1	1
	MM1	12	2	0	3
	MM2	12	2	0	3

Table 1. Leading terms in the complexities per iteration of the convolutive NMF algorithms (number of flop operations). Lower order terms are not shown for conciseness.

Interestingly, it can be shown that performing this update is equivalent to performing the update (12) sequentially, i.e., for $n = 1, \ldots, N$, without refreshment of $\hat{\mathbf{V}}$ after *each* update of $\mathbf{h}_n$ but only once after updating *all* the columns. This leads to an efficient sequential algorithm that performs a global MM update of $\mathbf{H}$, which avoids creating and manipulating the large-scale $MN \times KN$ matrix $\mathcal{W}$. This algorithm will be referred to as MM2 and is summarized in Algorithm 2.

3.5. Complexity analysis

The complexity of the three convolutive algorithms obtained by combining the update of $\{\mathbf{W}(t)\}_t$ defined in Section 3.1 with those of $\mathbf{H}$ proposed in Sections 3.2, 3.3 and 3.4 are given in Table 1. We can see that the dominant terms are $\mathcal{O}(MKNT)$ for all three algorithms and divergences, and hence T times the complexity $\mathcal{O}(MKN)$ of traditional NMF.[1]

4. EXPERIMENTS

4.1. Description of the data

The following experiments will use a 23-second musical excerpt of *Mamavatu* by Susheela Raman sampled at 16 kHz. We used a 640 point spectrum (40 ms) which resulted in $M = 321$ distinct frequency bins. We applied 50% overlapping sinebell windows before performing the discrete Fourier transform. This leads to $N = 1191$ time frames. We consider three values of $\beta \in \{0, 1, 2\}$, corresponding to the IS, KL and squared Euclidean loss, respectively. The nonnegative data matrix $\mathbf{V}$ is computed as the squared magnitude spectrogram for $\beta = 0$ and as the magnitude spectrogram for $\beta = 1$ and $\beta = 2$, which corresponds to three commonly used choices [2]. Each run the three convolutive NMF algorithms is initialized the same starting points. We perform experiments based on a fixed number of iterations N_{iter} which appears to be roughly equivalent to fixing the execution time of each algorithm. All the experiments are run on a single core of a DELL PowerEdge R620 equipped with 2 Intel Xeon E5-2690v2 @ 3.0 GHz and 128 GB of RAM. The experiments were run using MATLAB. Code implementing the algorithms and the experiments is made available online.[2]

[1] Note that complexity of traditional NMF can be reduced when $\beta = 2$ by computing $\mathbf{W}^T\hat{\mathbf{V}}$ as $(\mathbf{W}^T\mathbf{W})\mathbf{H}$ [13]. Unfortunately, this trick is not possible in convolutive NMF since $\hat{\mathbf{V}}$ is a sum of several terms in this case.

[2] https://www.irit.fr/~Cedric.Fevotte/extras/icassp2019/code.zip

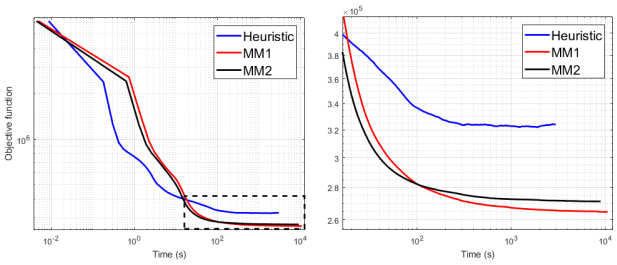

Fig. 1. Evolution of the objective function over 10^4 iterations for the *Mamavatu* experiment with the execution time (left) and zoom at the end (right).

T	Algorithm	$\beta = 0$	$\beta = 1$	$\beta = 2$
3	Heuristic	41	35	34
	MM1	360	302	149
	MM2	247	203	87
5	Heuristic	76	54	65
	MM1	523	321	220
	MM2	426	263	142
10	Heuristic	136	112	119
	MM1	1045	890	450
	MM2	860	611	292

Table 2. Execution time per iteration (in milliseconds).

4.2. Single run experiment

To start, we carry out an experiment where the algorithms are run once for $\beta = 0$ with $T = 10$, $K = 10$ and using $N_{\text{iter}} = 10^4$ iterations, providing an overall idea of the execution speeds and behavior of the objective function values. We can see in Fig. 1 that the heuristic runs faster than the two MM algorithms on this run, and that MM1 is slightly slower than MM2. The experiment highlights the fact that the heuristic does not always decrease the objective function value, while the proposed MM algorithms ensures its monotonic decrease. Moreover, MM1 and MM2 yield substantially smaller objective function values than the heuristic (and after a few iterations only).

4.3. Execution time

The second experiment uses the same setting than in Section 4.2 but considers multiple values for T and β. The goal is to examine the influence of these parameters on the execution speed. Results are reported in Table 2 and confirm that the execution time for each algorithm is proportional to (and hence increases with) convolution length T. This was to be expected given the linear complexity $\mathcal{O}(MKNT)$ per iteration of all algorithms. The algorithms have similar speed for $\beta = 1$ and $\beta = 2$, and run slower for $\beta = 0$. For each value of β, the two MM algorithms are found to be slower than the heuristic algorithm. While this cannot be explained from the complexity values reported in Table 2, which are similar for a given β, the observed slow-down is likely caused by the use of MATLAB for-loops in the update of $\mathbf{H}$ with the MM algorithms.

4.4. Objective function minimization

In a third experiment, we investigate the ability of each algorithm to minimize the divergence between the nonnegative data matrix $\mathbf{V}$ and its convolutive NMF approximation $\hat{\mathbf{V}}$. The algorithms are run for $N_{\text{iter}} = 1000$ iterations and the end values of the objective function are averaged over 100 random initializations. Results are reported in Table 3. They show that the two MM algorithms provide equivalent performances and perform significantly better than the heuristic in every case. MM1 appears to be slightly better than MM2. Interestingly, the objective function end value increases with T for the heuristic, while it decreases (as should be expected) for the MM algorithms. The MM algorithms can take advantage of the increasing degrees of freedom offered by larger values of T. In contrast, the heuristic on which the update (7-8) is based seems less and less plausible as T increases. Note that qualitatively similar results (not reproduced here due to space limitations) were obtained when comparing objective function values after a fixed budget of CPU time instead of iterations.

T	Algorithm	$\beta = 0$ ($\times 10^3$)	$\beta = 1$	$\beta = 2$
3	Heuristic	312.4 ± 4.2	1190.0 ± 13.7	134.8 ± 2.0
	MM1	288.5 ± 3.3	1125.8 ± 11.6	119.0 ± 2.5
	MM2	288.6 ± 3.2	1126.1 ± 11.2	119.3 ± 2.4
5	Heuristic	320.7 ± 5.1	1208.4 ± 14.4	147.0 ± 4.3
	MM1	281.6 ± 3.4	1099.2 ± 12.1	114.4 ± 2.0
	MM2	285.4 ± 3.3	1098.4 ± 11.4	115.3 ± 1.9
10	Heuristic	337.2 ± 11.3	1246.4 ± 34.3	211.2 ± 24.6
	MM1	271.5 ± 3.8	1054.9 ± 12.5	107.7 ± 2.0
	MM2	276.3 ± 3.7	1054.9 ± 14.2	108.0 ± 2.2

Table 3. Average objective function values (with standard deviations over 100 random initializations) reached after $N_{\text{iter}} = 1000$.

T	$\beta = 0$	$\beta = 1$	$\beta = 2$
3	11.6 ± 12.9	26.7 ± 24.3	24.6 ± 22.1
5	15.7 ± 17.8	21.0 ± 21.5	19.6 ± 13.4
10	2.1 ± 6.6	19.0 ± 24.0	39.5 ± 17.8

Table 4. Average percentage (with standard deviations over 100 random initializations) of times the objective function value has actually increased using the heuristic algorithm ($N_{\text{iter}} = 1000$ iterations).

Finally, Table 4 shows the fraction (in %) of iterations for which the heuristic algorithm has actually increased the objective function value between two consecutive iterations (remember that the MM algorithms ensure its decrease at each iteration). Up to 25% of the iterations can lead to an increase of the objective function for the scenarios considered here. This is coherent with the inferior performance in terms of objective function value obtained for the heuristic reported in Table 3. Indeed, any of these increases is bound to impact the overall minimization performances as observed in Fig. 1.

5. CONCLUSION

This paper addressed the problem of finding rigorous MM-based updates for convolutive NMF with the β-divergence. We proposed two rigorous algorithms that ensure monotonic decrease of the objective function. Experiments showed that the proposed MM algorithms consistently provide lower values of the objective function than the heuristic. All algorithms have similar complexity, yet the MM updates are found to be slightly slower because of implementation issues. Overall, the sequential algorithm MM2 for the global update of $\mathbf{H}$ yields best overall trade-off between computation time and performance.

6. REFERENCES

[1] Daniel D. Lee and H. Sebastian Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[2] Paris Smaragdis, Cédric Févotte, Gautham J Mysore, Nasser Mohammadiha, and Matthew Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 66–75, 2014.

[3] Emmanuel Vincent, Tuomas Virtanen, and Sharon Gannot, *Audio source separation and speech enhancement*, John Wiley & Sons, 2018.

[4] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-ichi Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, John Wiley & Sons, 2009.

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

[6] Emmanuel Vincent, Nancy Bertin, and Roland Badeau, "Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008.

[7] Paris Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Transactions on Audio Speech and Language Processing*, vol. 15, no. 1–12, pp. 1, 2007.

[8] Wenwu Wang, Andrzej Cichocki, and Jonathon A. Chambers, "A multiplicative algorithm for convolutive non-negative matrix factorization based on squared Euclidean distance," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2858–2864, 2009.

[9] David R. Hunter and Kenneth Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, 2004.

[10] Cédric Févotte and Jérôme Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[11] Andrzej Cichocki and Shun-ichi Amari, "Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities," *Entropy*, vol. 12, no. 6, pp. 1532–1568, 2010.

[12] Cédric Févotte and Nicolas Dobigeon, "Nonlinear hyperspectral unmixing with robust nonnegative matrix factorization," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4810–4819, 2015.

[13] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis," *Neural computation*, vol. 21, no. 3, pp. 793–830, 2009.