Block Bregman Majorization Minimization with Extrapolation*

Le Thi Khanh Hien†, Duy Nhat Phan‡, Nicolas Gillis†, Masoud Ahookhosh§, and Panagiotis Patrinos¶

Abstract. In this paper, we consider a class of nonsmooth nonconvex optimization problems whose objective is the sum of a block relative smooth function and a proper and lower semicontinuous block separable function. Although the analysis of block proximal gradient (BPG) methods for the class of block L-smooth functions have been successfully extended to Bregman BPG methods that deal with the class of block relative smooth functions, accelerated Bregman BPG methods are scarce and challenging to design. Taking our inspiration from Nesterov-type acceleration and the majorization-minimization scheme, we propose a block alternating Bregman Majorization-Minimization framework with Extrapolation (BMME). We prove subsequential convergence of BMME to a first-order stationary point under mild assumptions, and study its global convergence under stronger conditions. We illustrate the effectiveness of BMME on the penalized orthogonal nonnegative matrix factorization problem.

Key words. inertial block coordinate method, majorization minimization, Bregman surrogate function, acceleration by extrapolation, orthogonal nonnegative matrix factorization

AMS subject classifications. 90C26, 49M37, 65K05, 15A23, 15A83

1. Introduction.

In this paper, we consider the following nonsmooth nonconvex optimization problem

$$\begin{aligned} &\text{minimize}_{x=(x_1,\dots,x_m)} && F(x) := f(x) + \sum_{i=1}^{m} g_i(x_i) \\ &\text{subject to} && x_i \in \mathcal{X}_i \text{ for } i = 1,\dots,m, \end{aligned} \tag{1.1}$$

where $\mathcal{X}_i$ is a closed convex set of a finite dimensional real linear space $\mathbb{E}_i$ for $i \in [m] := \{1, 2, \dots, m\}$, x can be decomposed into m blocks $x = (x_1, \dots, x_m)$ with $x_i \in \mathcal{X}_i$, f is a continuously differentiable function, and g_i is a proper and lower semicontinuous function (possibly with extended values), and $\mathcal{X}_i \cap \mathbf{dom}\, g_i \neq \emptyset$. We denote $\mathcal{X} := \prod_{i=1}^{m} \mathcal{X}_i$. We assume F is bounded from below throughout the paper.

1.1. Related works.

The composite separable optimization problem (CSOP) (1.1) has been widely studied. It covers many applications including compressed sensing [7], sparse dictionary learning [1, 35], nonnegative tensor factorization [34, 19], and regularized sparse regression problems [11, 25]. When f has the block Lipschitz smooth property (that is, for

*LTK Hien and DN Phan contributed equally to this work.

Funding: The authors acknowledge the support by the European Research Council (ERC starting grant no 679515), and by the Fonds de la Recherche Scientifique - FNRS and the Fonds Wetenschappelijk Onderzoek - Vlaanderen (FWO) under EOS Project no O005318F-RG47.

†Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, Rue de Houdain 9, 7000 Mons, Belgium (thikhanhhien.le@umons.ac.be, nicolas.gillis@umons.ac.be).

‡Dynamic Decision Making Laboratory, Carnegie Mellon University, USA (dnphan@andrew.cmu.edu).

§Department of Mathematics, University of Antwerp, Belgium (masoud.ahookhosh@uantwerp.be).

¶Department of Electrical Engineering (ESAT-STADIUS), KU Leuven, Belgium (panos.patrinos@esat.kuleuven).

all $i \in [m]$ and fixing the values of x_j for $j \neq i$, the block function $x_i \mapsto f(x)$ admits an L_i-Lipschitz continuous gradient), then the nonconvex CSOP can be efficiently solved by block proximal gradient (BPG) methods [10, 13, 31, 33]. These methods update each block i, while fixing the value of blocks x_j for $j \neq i$, by minimizing over $x_i \in \mathcal{X}_i$ the block Lipschitz gradient surrogate function (see [20, Section 4]) as follows

$$x_i^{k+1} \in \operatorname*{argmin}_{x_i \in \mathcal{X}_i} \left\langle \nabla f_i^k(x_i^k), x_i - x_i^k \right\rangle + \frac{1}{2\gamma_i^k}\|x_i - x_i^k\|^2 + g_i(x_i), \tag{1.2}$$

where x_i^k is the value of block i at iteration k, $f_i^k(\cdot)$ denotes the value of the block function $x_i \to f(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \ldots, x_m^k)$, and γ_i^k is a step-size. To accelerate the BPG methods, several inertial versions have been proposed, including

(i) the heavy ball type acceleration methods in [28] that calculate an extrapolation point $\bar{x}_i^k = x_i^k + \beta_i^k(x_i^k - x_i^{k-1})$, then solving (1.2) by replacing the proximal term $\frac{1}{2\gamma_i^k}\|x_i - x_i^k\|^2$ by $\frac{1}{2\gamma_i^k}\|x_i - \bar{x}_i^k\|^2$,

(ii) the Nesterov-type acceleration methods in [34, 36] that takes the same step as the heavy ball acceleration but also replace $\nabla f_i^k(x_i^k)$ in (1.2) by $\nabla f_i^k(\bar{x}_i^k)$, and

(iii) the acceleration methods using two extrapolation points in [19, 29] that evaluate the gradient ∇f_i^k in (1.2) at an extrapolation point different from $\bar{x}_i^k$.

These methods were proved to have convergence guarantees when solving the nonconvex CSOP. The analysis of BPG methods has been extended to Bregman BPG methods [3, 18, 32] that replace the proximal term $\frac{1}{2}\|x_i - x_i^k\|^2$ in (1.2) by a Bregman divergence $D_{\psi_i}(x_i, x_i^k)$ associated with a kernel function ψ_i (see Definition 2.2) as follows

$$\min_{x_i \in \mathcal{X}_i} \left\langle \nabla f_i^k(x_i^k), x_i - x_i^k \right\rangle + \frac{1}{\gamma_i^k} D_{\psi_i}(x_i, x_i^k) + g_i(x_i). \tag{1.3}$$

The Bregman BPG methods can deal with a larger class of nonconvex CSOP in which the block function $x_i \mapsto f(x)$ may not have a L_i-Lipschitz continuous gradient, but is a relative smooth function (also known as a smooth adaptable function) [9, 22, 14]. Although the convergence analysis of BPG methods has been successfully extended to Bregman BPG methods, the convergence guarantees of their inertial versions for solving the nonconvex CSOP have not been studied much. In fact, to the best of our knowledge, there are only two papers addressing the convergence of inertial versions of Bregman BPG methods for solving (1.1), namely [2] and [20]. In [2], the authors consider an inertial Bregman BPG method that adds to $\nabla f_i^k(x_i^k)$ in (1.3) a *weak inertial force*, $\alpha_i^k(x_i^k - x_i^{prev})$, where α_i^k is some extrapolation parameter and x_i^{prev} is the previous value of x_i^k. In [20, Section 4.3], the authors introduce a heavy ball type acceleration with backtracking. The analysis of this method can be extended to a Nesterov-type acceleration with backtracking; however, the back-tracking procedure in [20, Section 4.3] for the Nesterov-type acceleration would be quite expensive since the computation of $f_i^k(\bar{x}_i^k)$ and $\nabla f_i^k(\bar{x}_i^k)$ would be required in the back-tracking process. Furthermore, there are no experiments in [2] and [20] to justify the efficacy of the inertial versions for Bregman BPG methods.

BPG and Bregman BPG methods belong to the block majorization minimization framework [20, 31] that updates one block x_i of x by minimizing a block surrogate function of

the objective function. In [20, Section 6.2], the matrix completion problem (MCP), which also has the form of Problem (1.1), illustrates the advantage of using suitable block surrogate functions and the efficacy of TITAN, the inertial block majorization minimization framework proposed in [20]. Specifically, each subproblem of TITAN that minimizes the block composite surrogate function[1] for F (which is formed by summing the Lipschitz gradient surrogate of f and a surrogate of g_i, see [20, Section 6.2]) has a closed-form solution while each proximal gradient step in the BPG method does not. Furthermore, TITAN outperforms BPG for the MCP. This motivates us to design an algorithm that allows using surrogate functions of g_i to replace g_i, in contrast to the current Bregman BPG methods which do not change g_i in the sub-problems; see (1.3).

1.2. Contribution and organization of the paper. After having introduced some preliminary notions of the Bregman distances and block relative smooth functions in Section 2, we propose in Section 3 a block alternating **B**regman **M**ajorization **M**inimization framework with **E**xtrapolation (BMME) that uses Nesterov-type acceleration to solve Problem (1.1) in which f is assumed to be a block relative smooth function with respect to $(\varphi_1, \ldots, \varphi_m)$; see Definition 2.4. This means that the gradient and the Bregman divergence in (1.3) are replaced with $\nabla f_i^k(\bar{x}_i^k)$ and $D_{\varphi_i^k}(x_i, \bar{x}_i^k)$, respectively; see Algorithm 3.1. We use a line-search strategy proposed in [24] to determine the extrapolation point $\bar{x}_i^k$. We remark that the inertial Bregman BPG method proposed in [24], named CoCaIn, is for solving the CSOP with $m = 1$ while BMME is for solving (1.1) with multiple blocks. Furthermore, CoCaIn requires its subproblem, which is Problem (1.3) with the gradient and the Bregman divergence being replaced by $\nabla f_i^k(\bar{x}_i^k)$ and $D_{\varphi_i^k}(x_i, \bar{x}_i^k)$ (note that we can omit the index i as $m = 1$ for CoCaIn), to be solved exactly (in other words, to have a closed-form solution). This requirement would be restrictive in applications where the nonsmooth part g_i is nonconvex and does not allow a closed form solution for the subproblem. In contrast, BMME employs surrogate functions for g_i, $i \in [m]$, that may lead to closed-form solutions for its subproblem, see an example in Section SM3. We note that CoCaIn requires $g_i(\cdot) + \alpha/2\|\cdot\|^2$ to be convex for some constant $\alpha \geq 0$ (see [24, Assumption C]) while BMME requires $x_i \mapsto u_i(x_i, y_i)$ to be convex for any $y_i \in \mathcal{X}_i$, where $u_i(\cdot, \cdot)$ is a surrogate function of g_i (see Definition 3.1). And as such, our analysis may allow a larger class of g_i than CoCaIn since u_i with $u_i(x_i, y_i) = g_i(x_i) + \alpha/2\|x_i - y_i\|^2$ is a surrogate function of g_i. It is important noting that the convexity assumption for the surrogate of g_i allows BMME to use stepsizes that only depend on the relative smooth constants of f. In contrast, CoCaIn needs to start with an initial relative smooth constant that linearly depends on the value of α that makes $g_i(\cdot) + \alpha/2\|\cdot\|^2$ convex. This initial relative smooth constant could be very large and lead to a very small stepsizes which results in a slow convergence. To illustrate this fact, we provide an experiment in Section SM3 to compare the performance of BMME and CoCaIn on the matrix completion problem.

In Section 4, we prove subsequential convergence of the sequence generated by BMME to a first-order stationary point of (1.1) under mild assumptions, and prove the global convergence under stronger conditions. Furthermore, the analysis in [24] does not consider the subse-

[1] It is worth noting that, in general, when TITAN uses Bregman surrogate functions for f, it does not change g_i in the subproblems, see [20, Section 4.3].

quential convergence but only proves the global convergence for F satisfying the Kurdyka-Łojasiewicz (KL) property [21], and under the assumption that the domains of the kernel functions are the full space. In our convergence analysis, we assume that every limit point x^* of the generated sequence by BMME satisfying the condition that x_i^* lies in the interior of the domain of $x_i \mapsto \varphi_i(x_1^*, \ldots, x_{i-1}^*, x_i, x_{i+1}^*, x_m^*)$ for $i \in [m]$. This assumption is naturally satisfied when the φ_i's have a full domain or $\mathcal{X} \subset \operatorname{int}\operatorname{dom}\varphi_i$. For example, the feasible set $\mathcal{X} = \{x : x_i \in \mathbb{R}^{d_i}, x_i \geq \varepsilon > 0\}$ (that is, each component of x_i is lower bounded by a positive constant ε) and the Burg entropy $\varphi_i(x) = -\sum_{j=1}^{d_i} \log x_{ij}$ satisfy our assumption; see for example the perturbed Kullback-Leibler nonnegative matrix factorization in [18]. We then prove subsequential convergence without the assumption that F satisfies the KL property, and prove global convergence (that is, the whole generated sequence converges from any feasible initial point) with this assumption.

In Section 5, we apply BMME to solve a penalized orthogonal nonnegative matrix factorization problem (ONMF). We conclude the paper in Section 6.

2. Preliminaries: Bregman distances and relative smoothness.

In this section, we present preliminaries of Bregman distances and relative smoothness. We adopt [14, Definition 2.1] to define a kernel generating distance which, for simplicity, we refer to as "kernel function".

Definition 2.1 (Kernel generating distance). *Let C be a nonempty, convex and open subset of $\mathbb{E}_i$. A function $\psi : \mathbb{E}_i \to \bar{\mathbb{R}} := (-\infty, +\infty]$ associated with C is called a kernel generating distance if it satisfies the following:*

(i) *ψ is proper, lower semicontinuous and convex with $\operatorname{dom}\psi \subset \bar{C}$, where $\bar{C}$ is the closure of C, and $\operatorname{dom}\partial\psi = C$.*

(ii) *ψ is continuously differentiable on $\operatorname{int}\operatorname{dom}\psi \equiv C$.*

Let us denote the class of kernel generating distances by $\mathcal{G}(C)$.

Definition 2.2. *Given $\psi \in \mathcal{G}(C)$, we define $D_\psi : \operatorname{dom}\psi \times \operatorname{int}\operatorname{dom}\psi \to \mathbb{R}_+$ as the* Bregman divergence *associated with the kernel function ψ as follows*

$$D_\psi(x_i, y_i) := \psi(x_i) - \psi(y_i) - \langle \nabla\psi(y_i), x_i - y_i \rangle .$$

Definition 2.3 ((L,l)-relative smooth function). *Given $\psi \in \mathcal{G}(C)$, let $\phi : \mathbb{E}_i \to (-\infty, +\infty]$ be a proper and lower semicontinuous function with $\operatorname{dom}\psi \subset \operatorname{dom}\phi$, which is continuously differentiable on $C = \operatorname{int}\operatorname{dom}\psi$. We say ϕ is (L,l)-relative smooth to ψ if there exist $L > 0$ and $l \geq 0$ such that for any $x_i, y_i \in C$,*

$$\phi(x_i) - \phi(y_i) - \langle \nabla\phi(y_i), x_i - y_i \rangle \leq L D_\psi(x_i, y_i), \tag{2.1}$$

and

$$-lD_\psi(x_i, y_i) \leq \phi(x_i) - \phi(y_i) - \langle \nabla\phi(y_i), x_i - y_i \rangle . \tag{2.2}$$

Whenever ϕ is convex, we may take $l = 0$ and Definition 2.3 recovers [22, Definition 1.1]. In the case $l = L$, Definition 2.3 recovers [14, Definition 2.2].

Given a function $f : \mathbb{E} \to (-\infty, +\infty]$, for each $i \in [m]$ and any fixed y_j for $j \neq i$, we define a block function $f(\cdot, y_{\neq i}) : \mathbb{E}_i \to (-\infty, +\infty]$ by

$$x_i \mapsto f(x_i, y_{\neq i}) := f(y_1, \ldots, y_{i-1}, x_i, y_{i+1}, \ldots, y_m). \tag{2.3}$$

Definition 2.4 (Block relative smooth function). *We say that $f : \mathbb{E} \to (-\infty, +\infty]$ is a block relative smooth function with respect to $(\varphi_1, \ldots, \varphi_m)$, where f is continuously differentiable on $\mathcal{C} = \operatorname{int}\operatorname{dom}\varphi_1 = \cdots = \operatorname{int}\operatorname{dom}\varphi_m$ and $\operatorname{dom}\varphi_1 = \ldots = \operatorname{dom}\varphi_m \subset \operatorname{dom} f$, if, for any $y \in \operatorname{dom}\varphi_i$ we have $\varphi_i(\cdot, y_{\neq i})$ is a kernel generating function and the function $f(\cdot, y_{\neq i})$ is a $(L_i^{y_{\neq i}}, l_i^{y_{\neq i}})$-relative smooth to $\varphi_i(\cdot, y_{\neq i})$, where $(L_i^{y_{\neq i}}, l_i^{y_{\neq i}})$ may depend on y_j, $j \neq i$.*

Throughout this paper we will assume the following.

Assumption 1. *We suppose $\mathcal{C} = \mathbf{int}\,\mathbf{dom}\,\varphi_1 = \cdots = \mathbf{int}\,\mathbf{dom}\,\varphi_m$, $\mathbf{dom}\,\varphi_1 = \ldots = \mathbf{dom}\,\varphi_m \subset \mathbf{dom}\, f$, $\mathcal{X} \cap \mathbf{dom}\,\varphi_1 \neq \emptyset$, the function f in (1.1) is a block relative smooth function with respect to $(\varphi_1, ..., \varphi_m)$.*

Let us make an important remark regarding Definition 2.4.

Flexibility of Definition 2.4.. Let us consider the notion of block relative smoothness in Definition 2.4 without $l_i^{y_{\neq i}}$, that is, the condition (2.2) is discarded. Similar definitions have been considered in [3] and [2]. In [3], the authors first define a multi-block kernel function $\psi : \mathbb{E}_1 \times \ldots \times E_m \to \bar{\mathbb{R}}$ [3, Definition 3.1], and then define multi-block relative smoothness of f with respect to this multi-block kernel function with the relative smooth constants $(L_1, \ldots, L_m)$ [3, Definition 3.4]. In [2], the authors define the block relative smoothness of f with respect to $(\psi_1, \ldots, \psi_m)$, where $\psi_i : \mathbb{E}_1 \times \ldots \mathbb{E}_m \to \bar{\mathbb{R}}$ is an i-th block kernel function [2, Definition 2.1], with the relative smooth constants $(L_1, \ldots, L_m)$ [2, Definition 2.2]. It is crucial to note that $L_1, \ldots, L_m$ in these definitions are *constants* and the stepsize used in the algorithms proposed in [3] and [2] to update each block i is *strictly less* than $1/L_i$. In contrast, our Definition 2.4 allows the block i relative smooth constant to change in the iterative process, that is, $L_i^{y_{\neq i}}$ and $l_i^{y_{\neq i}}$ are not constants but vary with respect to the values of the other blocks y_j for $j \neq i$. This flexibility in Definition 2.4 will lead to more flexible choices for the block kernel functions, and also leads to variable step-sizes in designing Bregman BPG algorithms for solving the multi-block CSOP. In fact, as we will see in Algorithm 3.1, the stepsize to update block i is $1/L_i^k$ which changes in the course of the iterative process. We will illustrate this crucial advantage of Algorithm 3.1 for solving the penalized ONMF problem in Section 5. Furthermore, it is important noting that if f satisfies [2, Definition 2.1] or [3, Definition 3.4], then f satisfies Definition 2.4 with the corresponding $L_i^{y_{\neq i}}$ being the constant L_i for all $i \in [m]$. However, the converse does not hold; see an example in Section 5.1. Hence Algorithm 3.1 applies to a broader class of problems, while allowing a more flexible choice of the step-sizes which will lead to faster convergence; see Section 5.2.

3. Block Alternating Majorization Minimization with Extrapolation.

Before introducing BMME, let us first recall the definition of a surrogate function as follows.

Definition 3.1. *A function $u_i : \mathcal{X}_i \times \mathcal{X}_i \to \bar{\mathbb{R}}$ is called a surrogate function of $g_i : \mathcal{X}_i \to \bar{\mathbb{R}}$ if the following conditions are satisfied:*

(a) *$u_i(y_i, y_i) = g_i(y_i)$ for all $y_i \in \mathcal{X}_i$,*

(b) *$u_i(x_i, y_i) \geq g_i(x_i)$ for all $x_i, y_i \in \mathcal{X}_i$.*

The approximation error is defined as $h_i(x_i, y_i) := u_i(x_i, y_i) - g_i(x_i)$.

For example, $u_i(x_i, y_i) = g_i(x_i) + \frac{\alpha}{2}\|x_i - y_i\|^2$, where α is a nonnegative constant, is always a surrogate function of g_i. In this case, $h_i(x_i, y_i) = \frac{\alpha}{2}\|x_i - y_i\|^2$. We refer the readers to [20, 31, 23] for more examples.

Denote $x^{k,0} = x^k$ and

$$x^{k,i} = (x_1^{k+1}, \ldots, x_i^{k+1}, x_{i+1}^k, \ldots, x_m^k).$$

For notation succinctness, we denote $\varphi_i^k(\cdot) := \varphi_i(\cdot, x_{\neq i}^{k,i-1})$, $L_i^k := L_i^{x_{\neq i}^{k,i-1}}$, and $l_i^k := l_i^{x_{\neq i}^{k,i-1}}$.

We can now introduce our BMME algorithm; see Algorithm 3.1. In particular, at iteration k, for each block i, BMME chooses a surrogate function u_i of g_i such that $x_i \mapsto u_i(x_i, y_i)$ is convex (as mentioned in the introduction, this condition is satisfied by the requirement that $g_i(\cdot) + \alpha/2\|\cdot\|^2$ is convex for some constant $\alpha \geq 0$ of [24]) and computes an extrapolated point $\bar{x}_i^k = x_i^k + \beta_i^k(x_i^k - x_i^{k-1}) \in \mathbf{int\ dom}\, \varphi_i^k$, where β_i^k is an extrapolation parameter satisfying

$$D_{\varphi_i^k}(x_i^k, \bar{x}_i^k) \leq \frac{\delta_i L_i^{k-1}}{L_i^k + l_i^k} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k),$$

for some $\delta_i \in (0,1)$. BMME then updates $x^{k,i}$ by

$$\begin{aligned} x_i^{k,i} &\in \operatorname*{argmin}_{x_i \in \mathcal{X}_i} \left\{ L_i^k D_{\varphi_i^k}(x_i, \bar{x}_i^k) + \left\langle \nabla_i f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i \right\rangle + u_i(x_i, x_i^k) \right\} \\ &= \operatorname*{argmin}_{x_i \in \mathbb{E}_i} \left\{ L_i^k D_{\varphi_i^k}(x_i, \bar{x}_i^k) + \left\langle \nabla_i f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i \right\rangle + \left(u_i(x_i, x_i^k) + I_{\mathcal{X}_i}(x_i)\right) \right\}, \end{aligned}$$

where $I_{\mathcal{X}_i}$ is the indicator function of $\mathcal{X}_i$. We make the following standard assumption for $\{x^k\}$, see for example [14, Assumption C]. Note that the initial points x^{-1} and x^0 are chosen in the interior domain of φ_i, $i \in [m]$.

Assumption 2. *We have $x^k \in \operatorname{int} \operatorname{dom} \varphi_i$, $i \in [m]$.*

Assumption 2 is naturally satisfied when the domain of φ_i is full space. See [14, Lemma 3.1] and [14, Remark 3.1] for a sufficient condition that ensures (3.2) to produce $x_i^{k+1} \in \mathbf{int\ dom}\, \varphi_i^k$, which implies that Assumption 2 holds.

Choice of the extrapolation parameters.. BMME needs to adequately choose the extrapolation parameters β_i^k's. Let us mention some special choices.

When $x_i \mapsto f(x_i, y_{\neq i})$ admits an $L_i^{y_{\neq i}}$-Lipschitz continuous gradient, that is, $\varphi(\cdot, y_{\neq i}) = \frac{1}{2}\|\cdot\|^2$, Condition (3.1) becomes

$$(\beta_i^k)^2 \|x_i^k - x_i^{k-1}\|^2 \leq \frac{\delta_i L_i^{k-1}}{L_i^k + l_i^k} \|x_i^k - x_i^{k-1}\|^2.$$

Therefore, we can choose any β_i^k such that $\beta_i^k \leq \sqrt{\frac{\delta_i L_i^{k-1}}{L_i^k + l_i^k}}$. Moreover, if $f(\cdot, y_{\neq i})$ is convex, we can take $l_i^k = 0$ and hence we can choose any $\beta_i^k \leq \sqrt{\frac{\delta_i L_i^{k-1}}{L_i^k}}$.

In general, [24, Lemma 4.2] showed that if the symmetry coefficient of φ_i^k, which is defined by $\inf\left\{ \frac{D_{\varphi_i^k}(x_i, y_i)}{D_{\varphi_i^k}(y_i, x_i)} : x_i, y_i \in \mathbf{int\ dom}\, \varphi_i^k \right\}$, is positive then, for a given

$$\kappa = \frac{\delta_i L_i^{k-1}}{L_i^k + l_i^k} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k) / D_{\varphi_i^k}(x_i^{k-1}, x_i^k) > 0,$$

Algorithm 3.1 BMME

1: Choose $x^{-1}, x^0 \in \operatorname{int}\operatorname{dom}\varphi_i$, $\delta_i \in (0,1)$, and set $k = 0$. Let u_i be a surrogate function of g_i such that $x_i \mapsto u_i(x_i, y_i)$ is convex for any $y_i \in \mathcal{X}_i$.

2: **repeat**

3: **for** $i = 1, ..., m$ **do**

4: Compute an extrapolation parameter β_i^k such that

$$D_{\varphi_i^k}(x_i^k, \bar{x}_i^k) \leq \frac{\delta_i L_i^{k-1}}{L_i^k + l_i^k} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k), \tag{3.1}$$

where $\bar{x}_i^k = x_i^k + \beta_i^k(x_i^k - x_i^{k-1}) \in \operatorname{int}\operatorname{dom}\varphi_i^k$.

5: Update $x^{k,i}$ by

$$x_i^{k+1} \in \operatorname*{argmin}_{x_i \in \mathcal{X}_i} \left\{ L_i^k D_{\varphi_i^k}(x_i, \bar{x}_i^k) + \left\langle \nabla_i f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i \right\rangle + u_i(x_i, x_i^k) \right\}. \tag{3.2}$$

6: **end for**

7: $k \leftarrow k + 1$.

8: **until** Stopping criterion.

there always exists $\gamma_i^k > 0$ such that the following condition is satisfied for all $\beta_i^k \in [0, \gamma_i^k]$

$$D_{\varphi_i^k}(x_i^k, \bar{x}_i^k) \leq \kappa D_{\varphi_i^k}(x_i^{k-1}, x_i^k), \tag{3.3}$$

which is equivalent to the condition (3.1). Therefore, β_i^k can be determined by a line search as follows. At each iteration, we initialize $\beta_i^k = \frac{\nu_i^{k-1}-1}{\nu_i^k}$, where $\nu_i^k = \frac{1}{2}\left(1 + \sqrt{1 + 4(\nu_i^{k-1})^2}\right)$ and $\nu_i^0 = 1$ as in Nesterov [26], and, while the inequality (3.1) does not hold, we decrease β_i^k by a constant factor $\eta_i \in (0,1)$, that is, $\beta_i^k \leftarrow \beta_i^k \eta_i$.

Before proceeding to the convergence analysis, we make an important remark: the relative smoothness constants L_i^k and l_i^k in Algorithm 3.1 are assumed to be known at the moment of updating x_i^k. In case these values are unknown (or their known lower/upper bounds are too loose), we can employ the convex-concave backtracking strategy as in the algorithm CoCaIn BPG proposed in [24, Section 3.1] to determine these values as well as the extrapolation parameter β_i^k. The upcoming convergence analysis would be similar in that case. In Section SM3, we consider the matrix completion problem (MCP) which has the form of Problem (1.1) with $m = 1$, and we illustrate BMME with the backtracking strategy on this problem when the values of the relative smooth constants are too small/large. The experiment presented in Section SM3 on the MCP shows the backtracking strategy significantly improves the performance of BMME, and also outperforms CoCaIn BPG. However, to simplify the presentation, we will only consider the convergence analysis of BMME for solving the multi-block Problem (1.1) when the relative smooth constants are assumed to be known.

4. Convergence analysis. In this section, we study the subsequential convergence of BMME under standard assumptions, and the global convergence under stronger conditions. For our upcoming analysis, we need the following first-order optimality condition of (1.1): x^* is a first-order stationary point of (1.1) if

$$\langle p(x^*), x - x^* \rangle \geq 0 \text{ for all } x \in \mathcal{X} \text{ and for some } p(x^*) \in \partial F(x^*). \tag{4.1}$$

As f is continuously differentiable as defined in Assumption 1, $\partial F(x^*) = \{\partial_{x_1} F(x^*)\} \times \ldots \times \{\partial_{x_m} F(x^*)\}$, where $\partial F(x^*)$ is the limiting-subdifferential of F at x^*, see Definition A.1 in Appendix A. Therefore, (4.1) is equivalent to

$$\langle p_i(x^*), x_i - x_i^* \rangle \geq 0 \text{ for all } x_i \in \mathcal{X}_i, \text{for some } p_i(x^*) \in \partial_{x_i} F(x^*) \text{ for } i \in [m]. \tag{4.2}$$

If x^* is in the interior of $\mathcal{X}$ or $\mathcal{X}_i = \mathbb{E}_i$ then (4.1) reduces to the condition $0 \in \partial F(x^*)$, that is, x^* is a critical point of F.

4.1. Subsequential convergence. The following theorem presents the subsequential convergence of the sequence generated by Algorithm 3.1 under an additional assumption on the surrogate function of g_i.

Assumption 3. *(A) For $i \in [m]$, the surrogate function $u_i(\cdot,\cdot)$ of g_i used in (3.2) in Algorithm 3.1 satisfies that $x_i \mapsto u_i(x_i, y_i)$ is convex.*
(B) For $i \in [m]$, $u_i(x_i, y_i)$ is continuous in y_i and lower semicontinuous in x_i.
(C) For $i \in [m]$, given $y_i \in \mathcal{X}_i$, there exists a function $x_i \mapsto \bar{h}_i(x_i, y_i)$ such that $\bar{h}_i(\cdot, y_i)$ is continuously differentiable at y_i and $\nabla_{x_i} \bar{h}_i(y_i, y_i) = 0$, and the approximation error $x_i \mapsto h_i(x_i, y_i) := u_i(x_i, y_i) - g_i(x_i)$ satisfies

$$h_i(x_i, y_i) \leq \bar{h}_i(x_i, y_i) \text{ for all } x_i \in \mathcal{X}_i. \tag{4.3}$$

For example, if $g_i(\cdot) + \frac{\alpha}{2}\|\cdot\|^2$ is convex for some constant $\alpha \geq 0$ then the surrogate $u_i(x_i, y_i) = g_i(x_i) + \frac{\alpha}{2}\|x_i - y_i\|^2$ satisfies Assumption 3 with $\bar{h}_i(x_i, y_i) = h_i(x_i, y_i) = \frac{\alpha}{2}\|x_i - y_i\|^2$. More examples can be found in [20].

Theorem 4.1. *Let $\{x^k\}$ be the sequence generated by Algorithm 3.1, and let Assumptions 1-3 be satisfied. The following statements hold.*
A) For $k = 0, 1, \ldots$ we have

$$F(x^{k,i}) \leq F(x^{k,i-1}) - L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + \delta_i L_i^{k-1} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k). \tag{4.4}$$

B) If there exists a positive number $\underline{L}$ such that[2] $\min_{k,i} L_i^k \geq \underline{L}$, *we have*

$$\sum_{k=0}^{+\infty} \sum_{i=1}^{m} D_{\varphi_i^k}(x_i^k, x_i^{k+1}) < +\infty. \tag{4.5}$$

[2]This is a standard assumption in analysing inertial block coordinate methods, see e.g., [34, Assumption 2], [36, Assumption 2], [19, Assumption 3] for similar assumptions when f is a block Lipschitz smooth function.

C) Assume that $\nabla_{x_i}\varphi_i(\cdot, y_{\neq i})$ for $i \in [m]$ is continuous in $y_{\neq i}$, $\{L_i^k\}$ for $i \in [m]$ and $\{x^k\}$ are bounded[3]*, and $\{\rho_i^k\}$ for $i \in [m]$ is bounded from below by $\rho > 0$, where ρ_i^k is the modulus of the strong convexity of φ_i^k. If x^* is a limit point of $\{x^k\}$ and*[4] *$x_i^* \in \operatorname{int}\operatorname{dom}\varphi_i(\cdot, x_{\neq i}^*)$, then x^* is a first-order stationary point of Problem (1.1).*

Proof. A) Since x_i^{k+1} is a solution to the convex problem (3.2), it follows from [27, Theorem 3.1.23] that for every $x_i \in \mathcal{X}_i$ we have

$$\left\langle L_i^k(\nabla\varphi_i^k(x_i^{k+1}) - \nabla\varphi_i^k(\bar{x}_i^k)) + \nabla f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i - x_i^{k+1}\right\rangle + u_i(x_i, x_i^k) \geq u_i(x_i^{k+1}, x_i^k). \tag{4.6}$$

By choosing $x_i = x_i^k$, we obtain

$$\left\langle L_i^k(\nabla\varphi_i^k(x_i^{k+1}) - \nabla\varphi_i^k(\bar{x}_i^k)) + \nabla f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i^k - x_i^{k+1}\right\rangle + u_i(x_i^k, x_i^k) \geq u_i(x_i^{k+1}, x_i^k). \tag{4.7}$$

Substituting $u_i(x_i^{k+1}, x_i^k) \geq g_i(x_i^{k+1})$ and $u_i(x_i^k, x_i^k) = g_i(x_i^k)$ into this inequality gives

$$\left\langle L_i^k(\nabla\varphi_i^k(x_i^{k+1}) - \nabla\varphi_i^k(\bar{x}_i^k)) + \nabla f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i^k - x_i^{k+1}\right\rangle + g_i(x_i^k) \geq g_i(x_i^{k+1}).$$

On the other hand, since f is a block relative smooth function, we have

$$f(x^{k,i}) \leq f(\bar{x}_i^k, x_{\neq i}^{k,i-1}) + \langle \nabla f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i^{k+1} - \bar{x}_i^k\rangle + L_i^k D_{\varphi_i^k}(x_i^{k+1}, \bar{x}_i^k),$$

and

$$f(\bar{x}_i^k, x_{\neq i}^{k,i-1}) + \langle \nabla f(\bar{x}_i^k, x_{\neq i}^{k,i-1}), x_i^k - \bar{x}_i^k\rangle \leq f(x^{k,i-1}) + l_i^k D_{\varphi_i^k}(x_i^k, \bar{x}_i^k),$$

By summing the three inequalities above, we obtain

$$F(x^{k,i}) \leq F(x^{k,i-1}) + L_i^k\left\langle \nabla\varphi_i^k(x_i^{k+1}) - \nabla\varphi_i^k(\bar{x}_i^k), x_i^k - x_i^{k+1}\right\rangle + L_i^k D_{\varphi_i^k}(x_i^{k+1}, \bar{x}_i^k) + l_i^k D_{\varphi_i^k}(x_i^k, \bar{x}_i^k). \tag{4.8}$$

Moreover, we have

$$L_i^k\left\langle \nabla\varphi_i^k(x_i^{k+1}) - \nabla\varphi_i^k(\bar{x}_i^k), x_i^k - x_i^{k+1}\right\rangle + L_i^k D_{\varphi_i^k}(x_i^{k+1}, \bar{x}_i^k) = -L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + L_i^k D_{\varphi_i^k}(x_i^k, \bar{x}_i^k), \tag{4.9}$$

[3] It follows from Inequality (4.4) that if F has bounded level sets then $\{x^k\}$ is bounded.

[4] As mentioned in the introduction, this condition is satisfied when φ_i has full domain or $\mathcal{X} \subset \mathbf{int}\,\mathbf{dom}\,\varphi_i$

Therefore, we obtain

$$\begin{aligned} F(x^{k,i}) &\leq F(x^{k,i-1}) - L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + \left(L_i^k + l_i^k\right) D_{\varphi_i^k}(x_i^k, \bar{x}_i^k) \\ &\leq F(x^{k,i-1}) - L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + \delta_i L_i^{k-1} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k), \end{aligned} \tag{4.10}$$

where the second inequality holds by (3.1). This implies A).

B) Summing (4.4) over $i = 1, .., m$ gives

$$F(x^{k+1}) \leq F(x^k) - \sum_{i=1}^m L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + \sum_{i=1}^m \delta_i L_i^{k-1} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k). \tag{4.11}$$

By summing up this inequality from $k = 0$ to $K - 1$, we obtain

$$\begin{aligned} F(x^K) + \sum_{i=1}^m \delta_i L_i^{K-1} D_{\varphi_i^{K-1}}(x_i^{K-1}, x_i^K) + \sum_{k=0}^{K-1} \sum_{i=1}^m (1-\delta_i) L_i^k D_{\varphi_i^k}(x_i^k, x_i^{k+1}) \\ \leq F(x^0) + \sum_{i=1}^m \delta L_i^{-1} D_{\varphi_i^{-1}}(x_i^{-1}, x_i^0), \end{aligned}$$

which gives the result.

C) Let x^* be a limit point of $\{x^k\}$. There exists a subsequence $\{x^{k_n}\}$ of $\{x^k\}$ converging to x^*. We have $D_{\varphi_i^k}(x_i^k, x_i^{k+1}) \geq \frac{\rho_i^k}{2}\|x_i^k - x_i^{k+1}\|^2$ since φ_i^k is ρ_i^k-strongly convex. Together with the assumption $\rho_i^k \geq \rho > 0$ and (4.5) we have $\|x^k - x^{k+1}\|$ converges to 0. Hence, $\{x^{k_n+1}\}$ and $\{x^{k_n-1}\}$ converge to x^*. Substituting $x_i = x_i^*$ and $k = k_n$ into (4.6) gives

$$\begin{aligned} \left\langle L_i^{k_n}(\nabla\varphi_i^{k_n}(x_i^{k_n+1}) - \nabla\varphi_i^{k_n}(\bar{x}_i^{k_n})) + \nabla f(\bar{x}_i^{k_n}, x_{\neq i}^{k_n,i-1}), x_i^* - x_i^{k_n+1} \right\rangle \\ + u_i(x_i^*, x_i^{k_n}) \geq u_i(x_i^{k_n+1}, x_i^{k_n}). \end{aligned} \tag{4.12}$$

By taking $n \to +\infty$, we have

$$\limsup_{n\to+\infty} u_i(x_i^{k_n+1}, x_i^{k_n}) \leq g_i(x_i^*), \tag{4.13}$$

where we have used the boundedness of $L_i^{k_n}$, the continuity of $u_i(x_i, \cdot)$, $\nabla\varphi_i$, and ∇f, $x_i^* \in \operatorname{int}\operatorname{dom}\varphi_i(\cdot, x_{\neq i}^*)$, and the fact that $x^{k_n+1} \to x^*$ as $n \to +\infty$. From this and the lower semi-continuity of $u_i(x_i, y_i)$, we have

$$\lim_{n\to+\infty} u_i(x_i^{k_n+1}, x_i^{k_n}) = g_i(x_i^*). \tag{4.14}$$

Choosing $k = k_n$ in (4.6) and letting $n \to +\infty$ implies that, for all $x_i \in \mathcal{X}_i$,

$$g_i(x_i^*) \leq u_i(x_i, x_i^*) + \left\langle \nabla f(x_i^*, x_{\neq i}^*), x_i - x_i^* \right\rangle. \tag{4.15}$$

Note that $u_i(x_i, x_i^*) = g_i(x_i) + h_i(x_i, x_i^*)$ and $f(\cdot, x_{\neq i}^*)$ is (L_i^*, l_i^*)-relative smooth to $\varphi_i^*(\cdot) = \varphi_i(\cdot, x_{\neq i}^*)$ for some constant L_i^*, l_i^*. Therefore, from (4.15) we have for all $x_i \in \mathcal{X}_i$ that

$$\begin{aligned} F(x^*) &\leq F(x_i, x_{\neq i}^*) + l_i^* D_{\varphi_i^*}(x_i, x_i^*) + h_i(x_i, x_i^*) \\ &\leq F(x_i, x_{\neq i}^*) + l_i^* D_{\varphi_i^*}(x_i, x_i^*) + \bar{h}_i(x_i, x_i^*), \end{aligned} \tag{4.16}$$

where $\bar{h}_i$ satisfies Assumption B (3). This implies that x_i^* is a minimizer of the following problem

$$\min_{x_i \in \mathcal{X}_i} F(x_i, x_{\neq i}^*) + l_i^* D_{\varphi_i^*}(x_i, x_i^*) + \bar{h}_i(x_i, x_i^*). \tag{4.17}$$

The result follows the optimality condition of (4.17) and $\nabla \bar{h}_i(x_i^*, x^*) = 0$. ∎

4.2. Global convergence. In order to prove the global convergence of Algorithm 3.1, we need to make an additional assumption.

Assumption 4. *For every iteration k of Algorithm 3.1, $f(\cdot, x_{\neq i}^{k,i-1})$ is relative smooth with respect to φ_i^k with constants (L_i^k, l_i^k) for $i \in [m]$. We will assume the following:*

(A) *There exist a positive integer number N, and $\underline{L}_i, \bar{L}_i > 0$ such that*

- $\underline{L}_i \leq \min_{k \geq N} L_i^k \leq \max_{k \geq N} L_i^k \leq \bar{L}_i$ *and* $\delta_i < \underline{L}_i / \bar{L}_i$;
- *for $i \in [m]$, φ_i^k is ρ_i^k- strongly convex and there exists $\rho > 0$ such that $\min_{k \geq N} \rho_i^k \geq \rho$.*

(B) *∇f and $\nabla \varphi_i$, for $i \in [m]$, are Lipschitz continuous on any bounded subsets of $\mathbb{E}$.*

We remark that Assumption 4 (A) on the boundedness of L_i^k is considered to be standard in the literature of inertial block coordinate methods, see [34, Assumption 2], [36, Assumption 2], [19, Assumption 3] for similar assumptions when considering block Lipschitz smooth problems. Assumptions 4 (B) is naturally satisfied when f and φ_i are twice continuously differentiable.

The global convergence of Algorithm 3.1 now can be stated for F satisfying the KL property, see Definition A.3 in Appendix A.

Theorem 4.2. *Assume that Assumptions 1 to 4 hold. Let $\{x^k\}$ be the sequence generated by Algorithm 3.1. We further assume that (i) $\{x^k\}$ is bounded, (ii) for any x_i, y_i in a bounded subset of $\mathcal{X}_i$ if $s_i \in \partial_{x_i}(I_{\mathcal{X}_i}(x_i) + u_i(x_i, y_i))$, there exists $\xi_i \in \partial(I_{\mathcal{X}_i}(x_i) + g_i(x_i))$ such that $\|\xi_i - s_i\| \leq A_i \|x_i - y_i\|$ for some constant[5] A_i, and (iii) F satisfies the KL property at any point $x^* \in \operatorname{dom} \partial \mathrm{F}$. Then the whole sequence $\{x^k\}$ converges to a critical point $\Phi(x) = F(x) + \sum_{i=1}^m I_{\mathcal{X}_i}(x_i)$.*

Proof. Consider the following auxiliary function

$$\begin{aligned} &\Phi^\gamma(x, y) = \\ &\Phi(x) + \sum_{i=1}^m \gamma_i D_{\varphi_i}\big((x_1, \ldots, x_{i-1}, y_i, \ldots, y_m), (x_1, \ldots, x_i, y_{i+1}, \ldots, y_m)\big) \end{aligned} \tag{4.18}$$

[5]This assumption is naturally satisfied if $u_i(x_i, y_i) = g_i(x_i)$ (that is, we use g_i itself as its surrogate). It is also satisfied if g_i and u_i are continuously differentiable, $\nabla_{x_i} u_i(x_i, x_i) = \nabla g_i(x_i)$, and $(x_i, y_i) \mapsto \nabla_{x_i} u_i(x_i, y_i)$ is Lipschitz continuous on any bounded subsets of $\mathcal{X}_i \times \mathcal{X}_i$ since we then have $\nabla x_i u_i(x_i, y_i) - \nabla g_i(x_i) = \nabla_{x_i}(u_i(x_i, y_i) - u_i(x_i, x_i))$, see [20] for some examples that satisfy these conditions.

where $\gamma_i = (\underline{L}_i + \delta_i \bar{L}_i)/2$, and let us denote $z^k = (x^k, x^{k-1})$. Then we have $\Phi^\gamma(z^k) = \Phi(x^k) + \sum_{i=1}^m \gamma_i D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k)$. Here we only need to prove that the sequence $\{z^k\}$ satisfies the three conditions **H**1, **H**2, and **H**3 in [8] since the result can be derived by using these conditions and the same arguments of the proof for [8, Theorem 2.9].

(**H**1) *Sufficient decrease condition.* It follows from (4.11) that for all $k \geq N+1$

$$\sum_{i=1}^m \underline{L}_i D_{\varphi_i^k}(x_i^k, x_i^{k+1}) + F(x^{k+1}) \leq F(x^k) + \sum_{i=1}^m \delta_i \bar{L}_i D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k). \tag{4.19}$$

Therefore, we have

$$\begin{aligned}
\Phi^\gamma(z^k) - \Phi^\gamma(z^{k+1}) &\geq \sum_{i=1}^m \frac{\underline{L}_i - \delta_i \bar{L}_i}{2} \left(D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k) + D_{\varphi_i^k}(x_i^k, x_i^{k+1}) \right) \\
&\geq \sum_{i=1}^m \frac{(\underline{L}_i - \delta_i \bar{L}_i)\rho_i}{4} \left(\|x_i^{k-1} - x_i^k\|^2 + \|x_i^k - x_i^{k+1}\|^2 \right) \\
&\geq \tau \|z^{k+1} - z^k\|^2,
\end{aligned}$$

where $\tau = \min_i(\underline{L}_i - \delta_i \bar{L}_i)\rho_i/4 > 0$ due to Assumption 4.

(**H**2) *Relative error condition.* By using the optimal condition of the subproblem (3.2) in BMME, we have for all $k \geq N+1$

$$s_i^{k+1} := L_i^k(\nabla \varphi_i^k(\bar{x}_i^k) - \nabla \varphi_i^k(x_i^{k+1})) - \nabla_i f(\bar{x}_i^k, x_{\neq i}^{k,i-1}) \in \partial(I_{\mathcal{X}_i}(x_i^{k+1}) + u_i(x_i^{k+1}, x_i^k)).$$

Hence, there exists $\xi_i^{k+1} \in \partial(I_{\mathcal{X}_i}(x_i^{k+1}) + g_i(x_i^{k+1}))$ such that

$$\|\xi_i^{k+1} - s_i^{k+1}\| \leq A_i \|x_i^{k+1} - x_i^k\|, \tag{4.20}$$

for some A_i. Therefore, we have $d_i^{k+1} := \nabla_{x_i} f(x^{k+1}) + \xi_i^{k+1} \in \partial_{x_i} \Phi(x^{k+1})$ and

$$\begin{aligned}
\left\| d_i^{k+1} \right\| = \left\| \nabla_{x_i} f(x^{k+1}) + s_i^{k+1} + \xi_i^{k+1} - s_i^{k+1} \right\| &\leq \left\| L_i^k(\nabla \varphi_i^k(\bar{x}_i^k) - \nabla \varphi_i^k(x_i^{k+1})) \right\| \\
&\quad + \left\| \nabla_{x_i} f(x^{k+1}) - \nabla_i f(\bar{x}_i^k, x_{\neq i}^{k,i-1}) \right\| + \left\| \xi_i^{k+1} - s_i^{k+1} \right\| \\
&\leq \left(\left(\bar{L}_i L^{\varphi_i} + L_i^f \right) \left(1 + \beta_i^k \right) + A_i \right) \left(\left\| x_i^{k+1} - x_i^k \right\| + \left\| x_i^k - x_i^{k-1} \right\| \right),
\end{aligned}$$

where the second inequality holds by (4.20), the boundedness of $\{x^k\}$, and the local Lipschitz continuity of $\nabla \varphi_i$ and ∇f. On the other hand, $\partial \Phi^\gamma(z^{k+1}) = \left(\partial_x \Phi^\gamma(z^{k+1}), \partial_y \Phi^\gamma(z^{k+1}) \right)$, where

$$\begin{aligned}
\partial_{x_i} \Phi^\gamma(z^{k+1}) = \partial_{x_i} \Phi(x^{k+1}) &+ \sum_{j=i+1}^m \gamma_j \left(\nabla_i \varphi_j(x^{k,j-1}) - \nabla_i \varphi_j(x^{k,j}) \right) \\
&- \sum_{j=i}^m \gamma_j \nabla_{ij}^2 \varphi_j(x^{k,j})(x_j^k - x_j^{k+1})
\end{aligned} \tag{4.21}$$

and

$$\partial_{y_i}\Phi^{\gamma}(z^{k+1}) = \sum_{j=1}^{i-1}\gamma_j\big(\nabla_i\varphi_j(x^{k,j-1}) - \nabla_i\varphi_j(x^{k,j})\big) - \sum_{j=1}^{i}\gamma_j\nabla^2_{ij}\varphi_j(x^{k,j})(x_j^k - x_j^{k+1}). \tag{4.22}$$

Therefore, we can deduce the relative error condition from the results above.

(**H3**) *Continuity condition.* Let x^* be a limit point of $\{x^k\}$. Since $\{x^k\}$ is bounded, there exists a subsequence $\{x^{k_n}\}$ of $\{x^k\}$ converging to x^*. Similarly to the proof of Theorem 4.1 (C), we can show that $u_i(x_i^{k_n}, x_i^{k_n-1}) \to g_i(x_i^*)$ as $n \to +\infty$. Therefore, we have

$$\begin{aligned}\limsup_{n\to+\infty} F(x^{k_n}) &\le \limsup_{n\to+\infty} f(x^{k_n}) + \sum_{i=1}^{m} u_i(x_i^{k_n}, x_i^{k_n-1})\\ &= f(x^*) + \sum_{i=1}^{m} g_i(x_i^*).\end{aligned} \tag{4.23}$$

On the other hand, since $\{\Phi^{\gamma}(x^k, x^{k-1})\}$ is non-increasing and bounded below, there exists $F^* = \lim_{k\to+\infty}\Phi^{\gamma}(x^k, x^{k-1})$. Moreover, $\lim_{k\to+\infty} D_{\varphi_i^{k-1}}(x_i^{k-1}, x_i^k) = 0$. This implies that $\lim_{k\to+\infty} F(x^k) = \lim_{k\to+\infty}\Phi^{\gamma}(x^k, x^{k-1}) = F^*$. By the uniqueness of the limit, we have $F^* = F(x^*)$.

By [8, Theorem 2.9] we have z^k converges to a critical point (x^*, x^*) of Φ^{γ}. Note that $\partial\Phi^{\gamma}(x^*, x^*) = (\partial\Phi(x^*), 0)$. The result follows then. ∎

Convergence rate.. We end this section by a remark on the convergence rate of BMME. By using the same arguments of the proof for [6, Theorem 2] we can derive a convergence rate for the generated sequence of BMME (see also [2, Theorem 3.14], [3, Theorem 4.7], [19, Theorem 3] and [34, Theorem 2.9]). We note that the convergence rate appears to be the same in different papers using the technique in [6]. Specifically, suppose $\mathbf{a}$ be a constant such that $\xi(s) = cs^{1-\mathbf{a}}$, where c is a constant, see Definition A.3. Then if $\mathbf{a} = 0$, BMME converges after a finite number of steps; if $\mathbf{a} \in (0, 1/2]$, BMME has linear convergence; and if $\mathbf{a} \in (1/2, 1)$, BMME has sublinear convergence. Determining the KŁ exponent $\mathbf{a}$ is out of the scope of this paper.

5. Numerical results.

In this section, we apply BMME to solve the following penalized orthogonal nonnegative matrix factorization (ONMF) [3]

$$\min_{U\in\mathbb{R}_+^{\mathbf{m}\times\mathbf{r}}, V\in\mathbb{R}_+^{\mathbf{r}\times\mathbf{n}}} f(U,V) := \tfrac{1}{2}\|X - UV\|_F^2 + \tfrac{\lambda}{2}\|I_r - VV^\top\|_F^2, \tag{5.1}$$

where $X \in \mathbb{R}_+^{\mathbf{m}\times\mathbf{n}}$ is a given input nonnegative data matrix and $\lambda > 0$ is a penalty parameter. ONMF is equivalent to a clustering problem. In fact, orthogonality ($VV^\top = I_r$) and nonnegativity of V implies that each column of V contains at most one non-zero entry; see [30] and the references therein. We implement all of the algorithms in MATLAB R2018a and run the experiments on a laptop with 1.8 GHz Intel Core i7 CPU and 16 GB RAM. The codes are available from https://github.com/LeThiKhanhHien/BMME.

5.1. Kernel functions and block updates of BMME. To implement BMME (Algorithm 3.1), we use the following kernel functions

$$\begin{aligned} \varphi_1(U,V) &= \frac{1}{2}\|U\|_F^2, \\ \varphi_2(U,V) &= \tfrac{6\lambda}{4}\|V\|_F^4 + \tfrac{1}{2}\varepsilon(U)\|V\|_F^2, \end{aligned} \tag{5.2}$$

where $\varepsilon(U) > 0$ may depend on U. Let us choose $\varepsilon(U) = \max\{\|U^\top U\|, 2\lambda\}$. Let us show that f is block relative smooth with respect to these kernel functions.

Proposition 5.1. *Fixing V, the function $f(\cdot, V)$ is $(L_1(V), l_1)$-relatively smooth with respect to $\varphi_1(\cdot, V)$, with $L_1(V) = \|VV^\top\|$ and $l_1 = 0$. Fixing U, $f(U,\cdot)$ is (L_2, l_2)- relatively smooth with respect to $\varphi_2(U,\cdot)$, with $L_2 = 1$ and $l_2 = 1$.*

Proof. The first statement is straightforward. Let us prove the second one. From [3, Proposition 5.1], we have

$$\nabla_V^2 f(U,V)[Z] = U^\top UZ + 2\lambda(ZV^\top V + VZ^\top V + VV^\top Z - Z).$$

Note that

$$\left|\left\langle ZV^\top V + VZ^\top V + VV^\top Z, Z\right\rangle\right| \le 3\|V\|_F^2\|Z\|_F^2. \tag{5.3}$$

Hence

$$\left\langle \nabla_V^2 f(U,V)[Z], Z\right\rangle \le \left\|U^\top U\right\| \|Z\|_F^2 + 6\lambda \|V\|_F^2 \|Z\|_F^2 .$$

Furthermore, we have

$$\nabla_V^2 \varphi_2(U,V)[Z] = 6\lambda\left(\|V\|_F^2 Z + 2\langle V, Z\rangle V\right) + \max\{\|U^\top U\|, \varepsilon\}Z,$$

which implies

$$\begin{aligned} L_2\left\langle \nabla_V^2\varphi_2(U,V)[Z], Z\right\rangle &= \max\{\left\|U^\top U\right\|, 2\lambda\}\|Z\|^2 + 6\lambda\left(\|V\|_F^2\|Z\|_F^2 + 2\langle V, Z\rangle^2\right) \\ &\ge \left\|U^\top U\right\|\|Z\|_F^2 + 6\lambda\|V\|_F^2\|Z\|_F^2 \ge \left\langle \nabla_V^2 f(U,V)[Z], Z\right\rangle. \end{aligned}$$

On the other hand, since $\max\{\|U^\top U\|, 2\lambda\} \ge 2\lambda$ we have

$$\begin{aligned} &\left\langle \nabla_V^2 f(U,V)[Z], Z\right\rangle + l_2\left\langle \nabla_V^2\varphi_2(U,V)[Z], Z\right\rangle \\ &\ge \left\langle U^\top UZ + 2\lambda(ZV^\top V + VZ^\top V + VV^\top Z), Z\right\rangle + 6\lambda\|V\|_F^2\|Z\|_F^2 \ge 0, \end{aligned}$$

where we have used (5.3) for the last inequality. The result follows, see [22, Proposition 1.1], [4, Proposition 2.6].

Proposition 5.1 shows that the kernel functions in (5.2) allow f to satisfy Definition 2.4, that is, f is block relative smooth with respect to these kernels. This would not hold for the block relative smoothness definitions from [2, Definition 2.2] and [3, Definition 3.4]. In fact, $L_1(V)$ depends on V so [2, Definition 2.2] does not apply, while φ_1 and φ_2 are two different functions so [3, Definition 3.4] does not apply either as it requires a sole multi-block kernel function to define block relative smoothness.

In the following we provide closed-form solutions of the sub-problems in (3.2) for the penalized ONMF problem.

Proposition 5.2. *Let φ_1 and φ_2 be defined in (5.2). Given $\bar{U}$, V, and L_1, we have*

$$\arg\min_{U\geq 0} \left\langle \nabla_U f(\bar{U},V), U\right\rangle + L_1 D_{\varphi_1(\cdot,V)}(U,\bar{U}) = \max\left(\bar{U} - \frac{1}{L_1}\left(\bar{U}VV^\top - XV^\top\right), 0\right).$$

Given $\bar{V}$, U, and L_2 we have

$$\arg\min_{V\geq 0} \left\langle \nabla_U f(U,\bar{V}), V\right\rangle + L_2 D_{\varphi_2(U,\cdot)}(V,\bar{V}) \;=\; \frac{1}{\rho}\max(G(\bar{V}),0),$$

where

$$\begin{aligned} G(\bar{V}) &= \nabla_V \varphi_2(U,\bar{V}) - \frac{1}{L_2}\nabla_V f(U,\bar{V}) \\ &= (6\lambda\|\bar{V}\|_F^2 + \varepsilon(U))\bar{V} - \frac{1}{L_2}\big(U^\top U\bar{V} - U^\top X + 2\lambda(\bar{V}\bar{V}^\top\bar{V} - \bar{V})\big), \end{aligned}$$

and ρ is the unique real solution of the equation $\rho^2(\rho - a) = c$, where $a = \varepsilon(U)$ and $c = 6\lambda\|\max(G(\bar{V}),0)\|^2$, so that ρ has the following closed form

$$\rho = \frac{a}{3} + \sqrt[3]{\frac{c+\sqrt{\Delta}}{2} + \frac{a^3}{27}} + \sqrt[3]{\frac{c-\sqrt{\Delta}}{2} + \frac{a^3}{27}},$$

where $\Delta = c^2 + \frac{4}{27}ca^3$.

Proof. For the update of U, we have

$$\begin{aligned} &\arg\min_{U\geq 0} \left\langle \nabla_U f(\bar{U},V), U\right\rangle + L_1 D_{\varphi_1(\cdot,V)}(U,\bar{U}) \\ &= \arg\min_{U\geq 0} \left\langle \nabla_U f(\bar{U},V), U\right\rangle + L_1\left(\varphi_1(U,V) - \varphi_1(\bar{U},V) - \left\langle \nabla_U\varphi_1(\bar{U},V), U-\bar{U}\right\rangle\right) \\ &= \arg\min_{U\geq 0} \varphi_1(U,V) - \left\langle \nabla_U\varphi_1(\bar{U},V) - \frac{1}{L_1}\nabla_U f(\bar{U},V), U\right\rangle \\ &= \max\left(\nabla_U\varphi_1(\bar{U}) - \frac{1}{L_1}\nabla_U f(\bar{U},V), 0\right) \\ &= \max\left(\bar{U} - \frac{1}{L_1}\left(\bar{U}VV^\top - XV^\top\right), 0\right). \end{aligned}$$

For the update of V, we have

$$\begin{aligned} &\arg\min_{V\geq 0} \left\langle \nabla_V f(U,\bar{V}), V\right\rangle + L_2 D_{\varphi_2(U,\cdot)}(V,\bar{V}) \\ &= \arg\min_{V\geq 0} \left\langle \nabla_V f(U,\bar{V}), V\right\rangle + L_2\left(\varphi_2(U,V) - \varphi_2(U,\bar{V}) - \left\langle \nabla_V\varphi_2(U,\bar{V}), V\right\rangle\right) \\ &= \arg\min_{V\geq 0} \varphi_2(U,V) - \left\langle G(\bar{V}), V\right\rangle. \end{aligned}$$

Using the same technique as in the proof of [3, Theorem 5.2], we obtain the result.

Computational cost of BMME for Penalized ONMF. The updates of BMME for (5.1) are given by Proposition 5.2. The main cost of the update of U is to compute $U(VV^\top)$ and $XV^\top$ which require $O((\mathbf{m}+\mathbf{n})\mathbf{r}^2)$ and $O(\mathbf{mnr})$ operations, respectively. Since $\mathbf{r} \ll \min(m,n)$, the update of U costs $O(\mathbf{mnr})$ operations, and is linear in the dimensions of the input matrix, as most NMF algorithms. The main cost of the update of V is to compute $(U^\top U)\bar{V}$, $U^\top X$, and $\bar{V}(\bar{V}^\top \bar{V})$ which require $O((\mathbf{m}+\mathbf{n})\mathbf{r}^2)$, $O(\mathbf{mnr})$ and $O(\mathbf{nr}^2)$ operations, respectively. Evaluating $D(\cdot,\cdot)$ in the backtracking line search to compute the extrapolation parameter costs $O(\mathbf{nr})$ operations. In summary, BMME requires $O(\mathbf{mnr})$ operations per iteration. Note that, if X is sparse, the cost per iteration reduces to $O(\text{nnz}(X)\mathbf{r})$ operations where $\text{nnz}(X)$ is the number of nonzero entries of X.

In summary, BMME has the same computational cost per iteration as most NMF algorithms, requiring $O(\text{nnz}(X)\mathbf{r})$ operations per iteration, the main cost being the computation of $U(VV^\top)$, $XV^\top$, $(U^\top U)V$, and $U^\top X$; see [15, Chapter 8] for a discussion.

Discussion on the assumptions for convergence. One can check that Assumptions 1, 2 and 3 are satisfied for Problem (5.1). Moreover, Equation (4.4) implies that $f(U^k, V^k)$ is upper bounded. Hence, to guarantee the boundedness of the generated sequence (as required in Theorem 4.1 (C) and Theorem 4.2), a possibility[6] is to lower bound the elements of U and V by a sufficiently small positive number, that is, $U_{ik} \geq \varepsilon$ and $V_{kj} \geq \varepsilon$ for some $\varepsilon > 0$. We recommend to set ε as the machine precision which we did in our implementation; see [15, Section 8.2.5] for a discussion. With this restriction, we also have $L_1(V) = \|VV^\top\| \geq \underline{L}$ for some positive number $\underline{L}$. On the other hand, $U \mapsto \varphi_1(U,V)$ is 1-strongly convex and $V \mapsto \varphi_2(U,V)$ is 2λ-strongly convex. Therefore, all the assumptions for Theorem 4.1 (C) are satisfied and as such BMME for solving Problem (5.1) guarantees a subsequential convergence. We note that φ_2 defined in (5.2) does not satisfy Assumption 4 (B), hence BMME using the kernel functions in (5.2) does not guarantee a global convergence for this problem.

5.2. Experiments on synthetic and real data sets. In the following, we compare the following algorithms on the penalized ONMF problem:

- BMME with the kernel functions defined in (5.2). At iteration k and for updating the blocks U and V, we find the extrapolation parameter β_i^k for BMME by starting from ν_k, where $\nu_0 = 1$ and $\nu_k = 1/2(1+\sqrt{1+4\nu_{k-1}})$ for $k \geq 1$, and then reducing it by a factor 0.9 until the condition (3.1) is satisfied.
- BMM, the non-extrapolated version of BMME.
- A-BPALM proposed in [3].
- BIBPA proposed in [2].

We provide the pseudo codes of A-BPALM and BIBPA in the supplementary material SM4. We use the kernel function $\varphi(U,V) = \frac{\alpha}{2}\|U\|_F^2\|V\|_F^2 + \frac{\beta}{4}\|V\|_F^4 + \frac{\varepsilon_1}{2}\|U\|_F^2 + \frac{\varepsilon_2}{2}\|V\|_F^2$, where $\alpha, \beta, \varepsilon_1$ and ε_2 are positive constants, for A-BPALM and BIBPA as proposed in [3, Proposition 5.1], and choose the default values for the parameters of A-BPALM and BIBPA in the upcoming experiments. All of these algorithms have convergence guarantee for solving the penalized ONMF problem (5.1), and have roughly the same computational cost per iteration, requiring $O(\text{nnz}(X)\mathbf{r})$ operations. We will display the evolution of the objective function

[6] Other strategies exist, for example adding an upper bound on the norm of the columns of U; see [15, Section 8.1.4] for a discussion.

values with respect to time; the evolution with respect to the iterations being very similar.

In the following four sections, we compare the four algorithms above on three types of data sets: synthetic data sets (Section 5.2.1), facial images (Section 5.2.2), and document data sets (Section 5.2.3).

5.2.1. Synthetic data sets. Let us compare the algorithms on synthetic data sets, as done in [3]. We use $(\mathbf{m}, \mathbf{n}, \mathbf{r}) = (500, 500, 10)$ and $(\mathbf{m}, \mathbf{n}, \mathbf{r}) = (500, 2000, 10)$. For each choice of $(\mathbf{m}, \mathbf{n}, \mathbf{r})$, we generate 30 synthetic data sets; each data set is generated as in. Specifically, we generate randomly the factor $U \in \mathbb{R}_+^{\mathbf{m}\times\mathbf{r}}$ using the MATLAB command rand. For V to have orthogonal rows (that is, $VV^\top = I$) and be nonnegative, V cannot have more than one non-zero entry per column. We generate an orthogonal nonnegative matrix $V \in \mathbb{R}_+^{\mathbf{r}\times\mathbf{n}}$ with a single nonzero entry in each column of V as follows. The position of the nonzero entry is picked at random[7] (with probability $1/r$ for each position), then the nonzero entry is generated using the uniform distribution in the interval $[0, 1]$, and finally we normalize each row of V. We construct a noiseless $X = UV$. Then we generate a noise matrix $R \in \mathbb{R}_+^{\mathbf{m}\times\mathbf{n}}$ using the MATLAB command rand. Finally we add 5% of noise, replacing the noiseless X as follows

$$X \leftarrow X + 0.05\frac{\|X\|_F}{\|R\|_F}R.$$

For each data set, we run each algorithm for 15 seconds and use the same initialization for all algorithms, namely the successive projection algorithm (SPA) [5, 17] as done in [3]. We set the penalty parameter $\lambda = 1000$ in our experiments. We report the evolution of the objective function with respect to time in Figure 1. We observe that BMME consistently outperforms the other algorithms in term of convergence speed, followed by BMM and A-BPALM. Note that the results are very consistent among various runs on different input matrices.

These experiments illustrate two facts:

1. Using extrapolation in BMME is useful and accelerates the convergence, as BMME outperforms BMM.
2. The flexibility of Definition 2.4 allows BMME to choose the kernel functions in (5.2) that also leads to a significant speedup as BMME outperforms A-BPALM and BIBPA. This illustrates our arguments in the paragraph "Flexibility of Definition 2.4" at the end of Section 2 (page 5).

In the next sections, we perform numerical experiments on real data sets to further validate these two key observations.

5.2.2. Facial images. In the second experiment, we compare the algorithms on four facial image data sets widely used in the NMF community: CBCL[8] (2429 images of dimension 19×19), Frey[9] (1965 images of dimension 28×20), ORL[10] (400 images of dimension 92×112), and Umist[11] (575 images of dimension 92×112). We construct X as an image-by-pixel matrix,

[7] Two non-zero entries of V in the same position in two different columns means that the two corresponding data points belong to the same cluster. In expectation, each row of V will have $\mathbf{n}/\mathbf{r}$ non-zero entries (that is, there are $\mathbf{n}/\mathbf{r}$ data points per cluster, in expectation).

[8] http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html

[9] https://cs.nyu.edu/~roweis/data.html

[10] https://cam-orl.co.uk/facedatabase.html

[11] https://cs.nyu.edu/~roweis/data.html

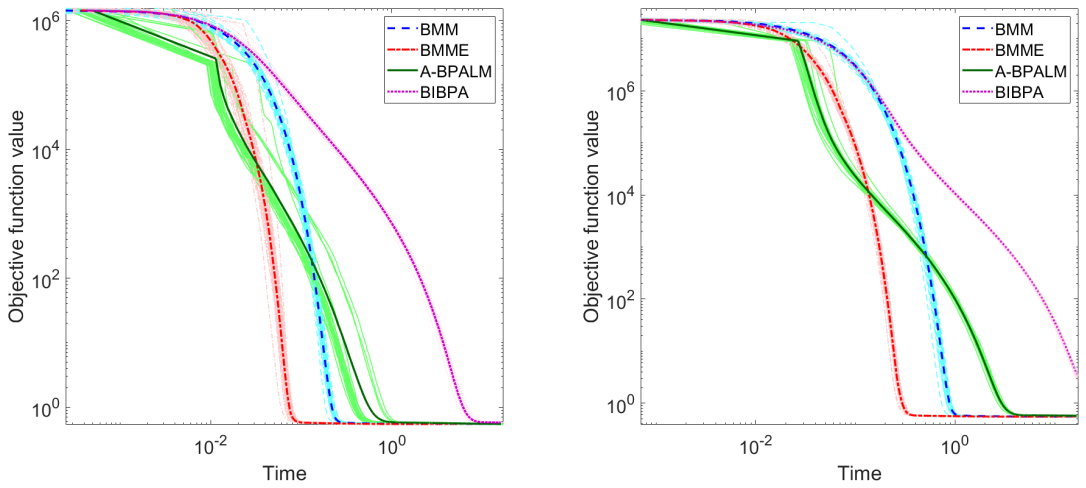

Figure 1. *Evolution of the objective functions with respect to the running time for 30 synthetic data sets with* $(\mathbf{m}, \mathbf{n}, \mathbf{r}) = (500, 500, 10)$ *(left) and* $(\mathbf{m}, \mathbf{n}, \mathbf{r}) = (500, 2000, 10)$ *(right), in a log-log scale. The average curve is plotted in bold.*

that is, each row of X is a vectorized facial image. As we will see, this allows ONMF to extract disjoint facial features as the rows of V. We set $\mathbf{r} = 25$ and use SPA initialization in all runs. We choose the penalty parameter $\lambda = \|X - U_0V_0\|_F^2/\mathbf{r}$, where (U_0, V_0) is the SPA initialization. We run each algorithm 100 seconds for each data set. The evolution of the scaled objective function values, which equal the objective function values divided by $\|X\|_F^2$, with respect to time is reported in Figure 2.

We observe a similar behavior as for the synthetic data sets: A-BPALM decreases the objective faster than BMME during the first milliseconds for these dense data sets[12] but, after about a hundredth of a second, BMME converges the fastest, followed by BMM. Figures 3 and 4 display the reshaped rows of V, corresponding to facial features, obtained by the different algorithms for the CBCL and ORL data sets, respectively. For the other data sets, Frey and Umist; see Section SM1.

In Figure 3, we observe that the solutions obtained by the 4 algorithms are similar. Because BIBPA did not have time to converge (see Figure 2), it generates slightly worse facial features, with some isolated pixels, and edges of the facial features being sharper.

In Figure 4, as BMM and BMME both converged to similar objective function values (see Figure 2), they provide very similar facial features; although slightly different. For example, the first facial feature of BMME is sparser than that of BMM. A-BPALM and BIBPA were not able to converge within the 100 seconds, and hence provide worse facial features. For example, the first facial feature is much denser than for BMM and BMME, overlapping with other facial features (meaning that the orthogonality constraints is not well satisfied). (A very similar observation holds for the Umist data set; see Section SM1.)

[12]This does not happen for the sparse document data sets la1, tr11, tr23, tr41 and tr45, but it happens for the other document data sets (the experiments for these data sets can be found in the supplementary material).

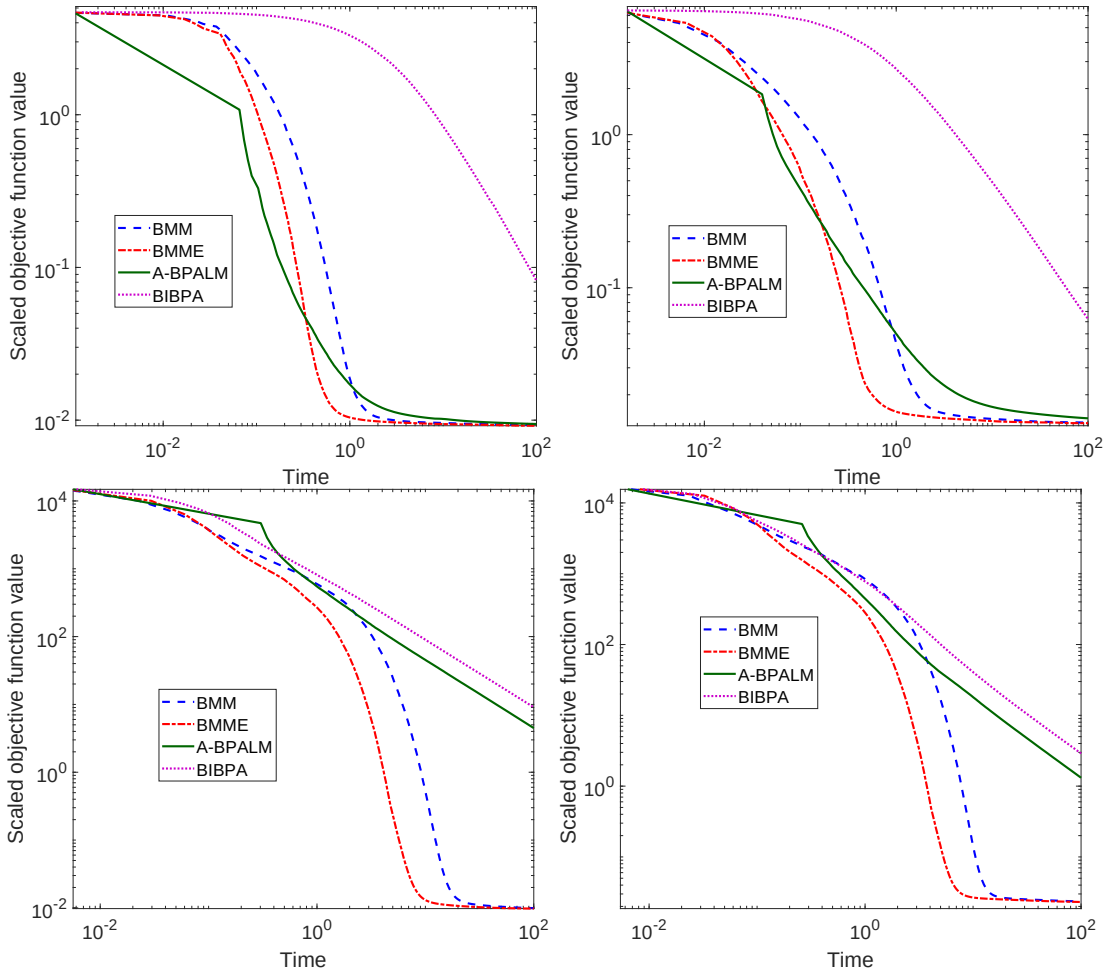

Figure 2. *Evolution of the scaled objective function values with respect to running time on the CBCL (top left), Frey (top right), Umist (bottom left) and ORL (bottom right) data sets, in loglog scale.*

5.2.3. Document data sets. In the third experiment, we compare the algorithms on 12 sparse document data sets from [37], as in [30]. For such data sets, SPA does not provide a good initialization, because of outliers and gross corruptions. Hence we initialize U_0 with the procedure provided by H2NMF from [16], while V_0 is initialized by minimizing $\|X - U_0V_0\|_F^2$ while imposing V_0 to have a single nonzero entry per column. The penalty parameter λ is chosen as before, namely $\lambda = \|X - U_0V_0\|_F^2/\mathbf{r}$. We run each algorithm 200 seconds for each data set. Table 1 reports the clustering accuracy obtained by the algorithms, which is defined as follows. Given the true clusters, C_k for $k = 1, 2, \ldots, \mathbf{r}$, and the clusters computed by an algorithm, C'_k for $k = 1, 2, \ldots, \mathbf{r}$ (in ONMF, a data point is assigned to the cluster corresponding to the largest entry in the corresponding column of V), the accuracy of the

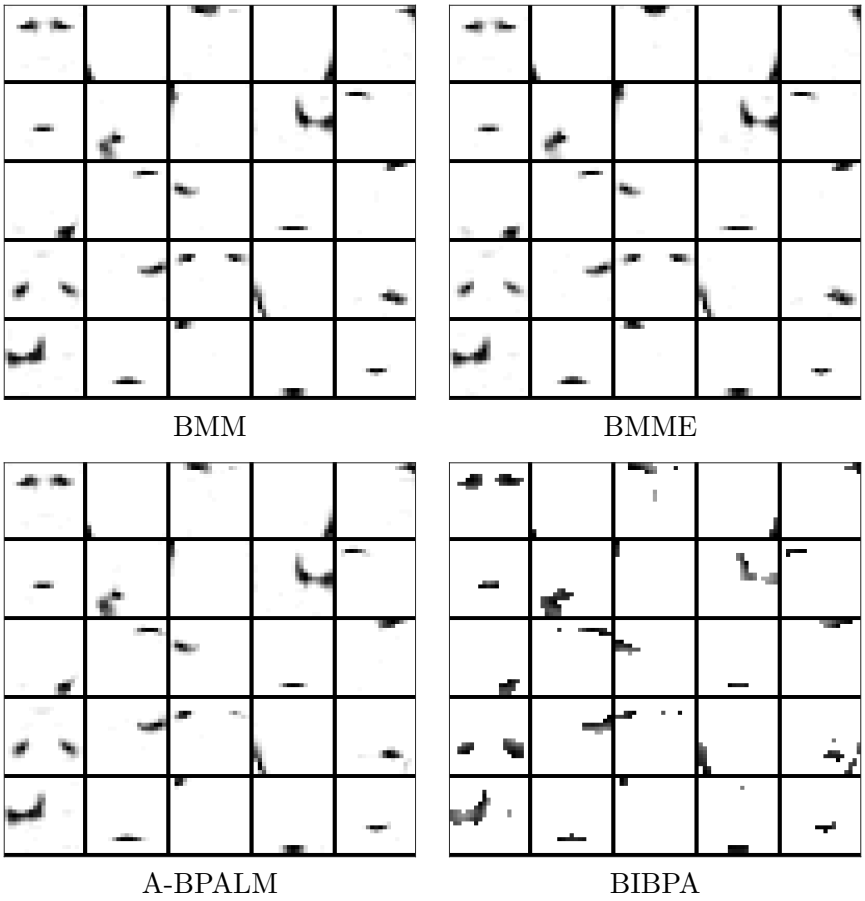

Figure 3. *Display of the rows of the matrix V as facial features, computed by BMM, BMME, A-BPALM, and BIBPA on the CBCL facial images.*

algorithm is defined as

$$\text{Accuracy} \quad = \max_{\pi,\ \text{a permutation of } [\mathbf{r}]} \frac{1}{\mathbf{n}} \left(\sum_{k=1}^{\mathbf{r}} \left| C_k \cap C'_{\pi(k)} \right| \right).$$

We observe on Table 1 that BMM and BMME provide, on average, better clustering accuracies than A-BPALM and BIBPA. In fact, in terms of accuracy, BMM performs similarly as BMME as both algorithms were able to converge within the allotted time (see Figure 5 and Section SM2). When A-BPALM or BIBPA have a better clustering accuracy, it is only by a small margin (less than 4% in all cases), while BMM and/or BMME sometimes outperform A-BPALM and BIBPA; in particular, by 6.8% for reviews, 7.2% for sports, 12% for la1, 10.7% for classic, and 7.8% for k1b. Interestingly, BMME and BMM actually return the same solutions when they have the same accuracy.

The scaled objective function values with respect to time for the hitech and reviews data set are reported in Figure 5; the results for the other data sets are similar, and can be found

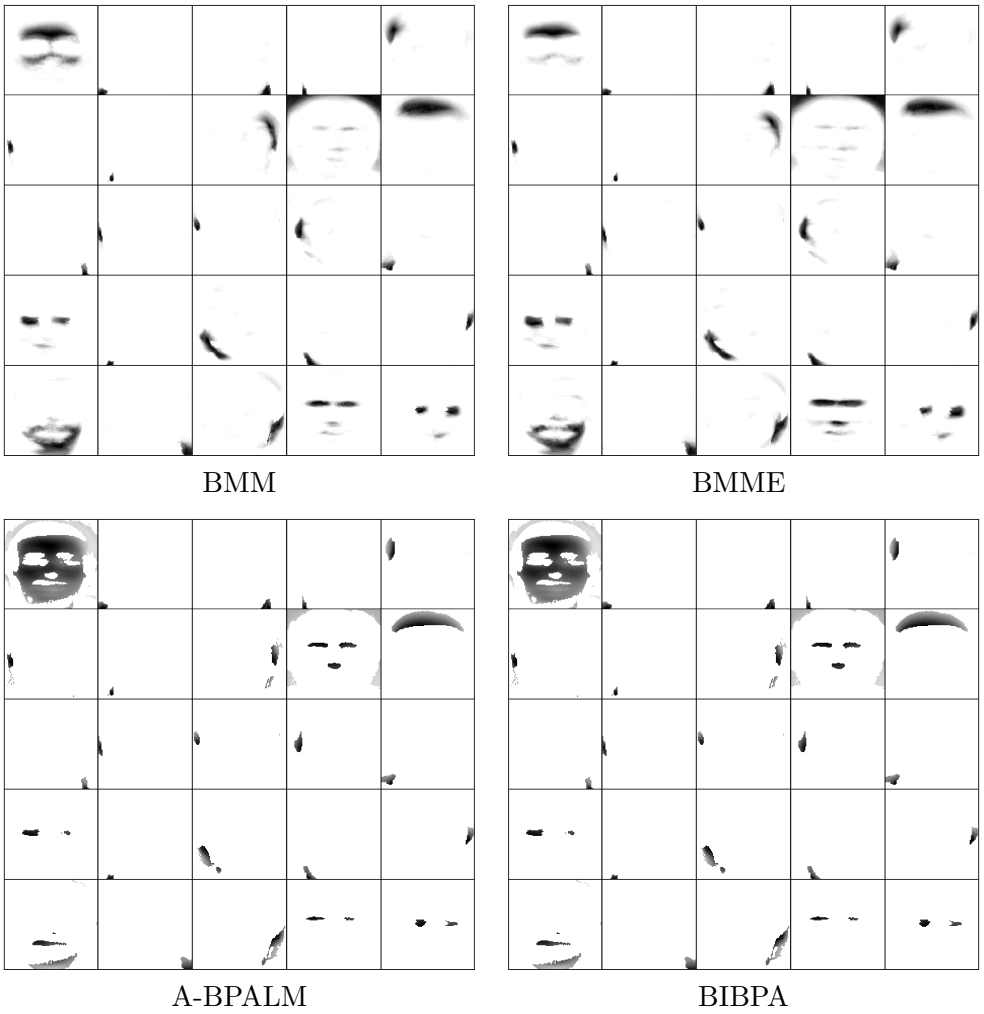

Figure 4. *Display of the rows of the matrix V as facial features, computed by BMM, BMME, A-BPALM, and BIBPA on the ORL facial images.*

in Section SM2. As before, BMME is the fastest, followed by BMM, A-BPALM and BIBPA (in that order).

6. Conclusion. In this paper, we have developed BMME, a block alternating Bregman Majorization Minimization framework with Extrapolation that uses the Nesterov acceleration technique, for a class of nonsmooth nonconvex optimization problems that does not require the global Lipschitz gradient continuity. We have proved the subsequential and global convergence of BMME to first-order stationary points; see Theorems 4.1 and 4.2, respectively. We have evaluated the performance of BMME on the penalized orthogonal nonnegative matrix factorization problem on synthetic data sets, facial images, and documents. The numerical results have shown that (1) Using extrapolation improves the convergence of BMME, and (2) BMME converges faster than previously introduced the Bregman BPG methods, A-

Table 1

Accuracy in percent obtained by the different algorithms on 12 document data sets. The best accuracy is highlighted in bold.

Data set	rank **r**	BMM	BMME	A-BPALM	BIBPA
hitech	6	**39.94**	39.93	38.98	37.07
reviews	5	**73.56**	73.53	66.70	66.31
sports	7	50.09	**50.13**	42.93	42.93
ohscal	10	**31.70**	31.52	27.25	27.25
la1	6	49.86	**53.37**	41.32	41.32
la2	6	53.43	52.46	**54.83**	50.34
classic	4	60.74	**61.43**	50.70	50.10
k1b	6	**79.19**	**79.19**	71.41	71.41
tr11	9	37.44	37.44	37.44	**41.30**
tr23	6	**41.67**	**41.67**	**41.67**	40.20
tr41	10	**38.61**	**38.61**	**38.61**	35.08
tr45	10	35.51	35.51	35.51	**37.82**
average		49.31	**49.57**	45.61	45.09

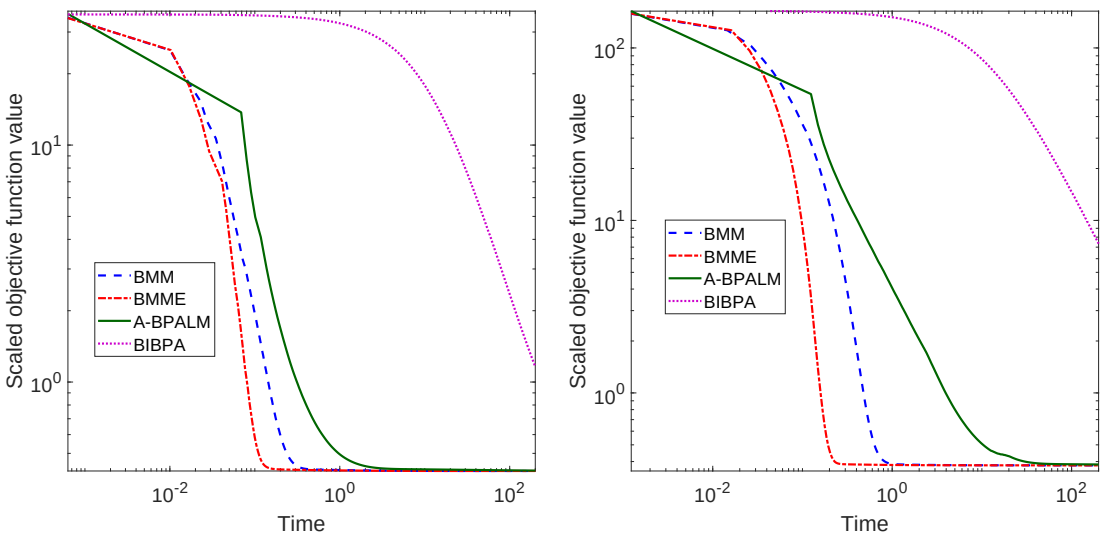

Figure 5. *Evolution of the scaled objective function values with respect to running time on the hitech (left) and review (right) data sets, in loglog scale.*

BPALM [3] and BIBPA [2], because BMME allows a much more flexible choice of the kernel functions and uses Nesterov-type extrapolation. We end the paper by an interesting question that we consider as a future research topic: can the cyclic update rule of BMME be extended to a randomized/non-cyclic update rule?

Acknowledgments. We thank the anonymous reviewers for their insightful comments that helped us improve the paper.

Appendix A. Preliminaries of nonconvex nonsmooth optimization.

Let $g : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous function.

Definition A.1. *(i) For each $x \in \operatorname{dom} g$, we denote $\hat{\partial} g(x)$ as the Frechet subdifferential of g at x which contains vectors $v \in \mathbb{E}$ satisfying*

$$\liminf_{y \neq x, y \to x} \frac{1}{\|y - x\|} \left(g(y) - g(x) - \langle v, y - x \rangle\right) \geq 0.$$

If $x \notin \operatorname{dom} g$, then we set $\hat{\partial} g(x) = \emptyset$.

(ii) The limiting-subdifferential $\partial g(x)$ of g at $x \in \operatorname{dom} g$ is defined as follows.

$$\partial g(x) := \left\{ v \in \mathbb{E} : \exists x^k \to x,\ g\left(x^k\right) \to g(x),\ v^k \in \hat{\partial} g\left(x^k\right),\ v^k \to v \right\}.$$

Definition A.2. *We call $x^* \in \operatorname{dom} \mathrm{F}$ a critical point of F if $0 \in \partial F\left(x^*\right)$.*

We note that if x^* is a local minimizer of F then x^* is a critical point of F.

Definition A.3. *A function $\phi(x)$ is said to have the KL property at $\bar{x} \in \operatorname{dom} \partial \phi$ if there exists $\eta \in (0, +\infty]$, a neighborhood U of $\bar{x}$ and a concave function $\xi : [0, \eta) \to \mathbb{R}_+$ that is continuously differentiable on $(0, \eta)$, continuous at 0, $\xi(0) = 0$, and $\xi'(s) > 0$ for all $s \in (0, \eta)$, such that for all $x \in U \cap [\phi(\bar{x}) < \phi(x) < \phi(\bar{x}) + \eta]$, we have*

$$\xi'\left(\phi(x) - \phi(\bar{x})\right) \operatorname{dist}\left(0, \partial \phi(x)\right) \geq 1. \tag{A.1}$$

$\operatorname{dist}\left(0, \partial \phi(x)\right) = \min\{\|y\| : y \in \partial \phi(x)\}$. *If $\phi(x)$ has the KL property at each point of $\operatorname{dom} \partial \phi$ then ϕ is a KL function.*

Many nonconvex nonsmooth functions in practical applications belong to the class of KL functions, for examples, real analytic functions, semi-algebraic functions, and locally strongly convex functions [12, 13].

REFERENCES

[1] M. Aharon, M. Elad, A. Bruckstein, et al., *K-svd: An algorithm for designing overcomplete dictionaries for sparse representation*, IEEE Transactions on Signal Processing, 54 (2006), p. 4311.

[2] M. Ahookhosh, L. T. K. Hien, N. Gillis, and P. Patrinos, *A block inertial Bregman proximal algorithm for nonsmooth nonconvex problems with application to symmetric nonnegative matrix tri-factorization*, Journal of Optimization Theory and Applications, (2021).

[3] M. Ahookhosh, L. T. K. Hien, N. Gillis, and P. Patrinos, *Multi-block Bregman proximal alternating linearized minimization and its application to sparse orthogonal nonnegative matrix factorization*, Computational Optimization and Application, 79 (2021), p. 681715.

[4] M. Ahookhosh, A. Themelis, and P. Patrinos, *Bregman forward-backward splitting for nonconvex composite optimization: superlinear convergence to nonisolated critical points*, SIAM Journal on Optimization, 31 (2021), pp. 653–685.

[5] U. Araújo, B. Saldanha, R. Galvão, T. Yoneyama, H. Chame, and V. Visani, *The successive projections algorithm for variable selection in spectroscopic multicomponent analysis*, Chemometrics and Intelligent Laboratory Systems, 57 (2001), pp. 65–73.

[6] H. Attouch and J. Bolte, *On the convergence of the proximal algorithm for nonsmooth functions involving analytic features*, Mathematical Programming, 116 (2009), pp. 5–16, https://doi.org/10.1007/s10107-007-0133-5.

[7] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, *Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality*, Mathematics of Operations Research, 35 (2010), pp. 438–457, https://doi.org/10.1287/moor.1100.0449.

[8] H. Attouch, J. Bolte, and B. F. Svaiter, *Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods*, Mathematical Programming, 137 (2013), pp. 91–129.

[9] H. H. Bauschke, J. Bolte, and M. Teboulle, *A descent lemma beyond Lipschitz gradient continuity: First-order methods revisited and applications*, Mathematics of Operations Research, 42 (2017), pp. 330–348, https://doi.org/10.1287/moor.2016.0817.

[10] A. Beck and L. Tetruashvili, *On the convergence of block coordinate descent type methods*, SIAM Journal on Optimization, 23 (2013), pp. 2037–2060.

[11] T. Blumensath and M. E. Davies, *Iterative hard thresholding for compressed sensing*, Applied and Computational Harmonic Analysis, 27 (2009), pp. 265 – 274, https://doi.org/10.1016/j.acha.2009.04.002.

[12] J. Bochnak, M. Coste, and M.-F. Roy, *Real Algebraic Geometry*, Springer, 1998.

[13] J. Bolte, S. Sabach, and M. Teboulle, *Proximal alternating linearized minimization for nonconvex and nonsmooth problems*, Mathematical Programming, 146 (2014), pp. 459–494.

[14] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd, *First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems*, SIAM Journal on Optimization, 28 (2018), pp. 2131–2151, https://doi.org/10.1137/17M1138558.

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

[16] N. Gillis, D. Kuang, and H. Park, *Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization*, IEEE Transactions on Geoscience and Remote Sensing, 53 (2015), pp. 2066–2078.

[17] N. Gillis and S. A. Vavasis, *Fast and robust recursive algorithms for separable nonnegative matrix factorization*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 36 (2013), pp. 698–714.

[18] L. T. K. Hien and N. Gillis, *Algorithms for nonnegative matrix factorization with the Kullback-Leibler divergence*, Journal of Scientific Computing, (2021), https://doi.org/10.1007/s10915-021-01504-0.

[19] L. T. K. Hien, N. Gillis, and P. Patrinos, *Inertial block proximal method for non-convex non-smooth optimization*, in Thirty-seventh International Conference on Machine Learning (ICML), 2020.

[20] L. T. K. Hien, D. N. Phan, and N. Gillis, *Inertial block majorization minimization framework for nonconvex nonsmooth optimization*. arXiv:2010.12133, 2020.

[21] K. Kurdyka, *On gradients of functions definable in o-minimal structures*, Annales de l'Institut Fourier, 48 (1998), pp. 769–783, https://doi.org/10.5802/aif.1638.

[22] H. Lu, R. M. Freund, and Y. Nesterov, *Relatively smooth convex optimization by first-order methods, and applications*, SIAM Journal on Optimization, 28 (2018), pp. 333–354.

[23] J. Mairal, *Optimization with first-order surrogate functions*, in Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ICML13, JMLR.org, 2013, pp. 783–791.

[24] M. C. Mukkamala, P. Ochs, T. Pock, and S. Sabach, *Convex-concave backtracking for inertial Bregman proximal gradient algorithms in nonconvex optimization*, SIAM Journal on Mathematics of Data Science, 2 (2020), pp. 658–682, https://doi.org/10.1137/19M1298007.

[25] B. Natarajan, *Sparse approximate solutions to linear systems*, SIAM Journal on Computing, 24 (1995), pp. 227–234, https://doi.org/10.1137/S0097539792240406.

[26] Y. Nesterov, *A method of solving a convex programming problem with convergence rate $O(1/k^2)$*, Soviet Mathematics Doklady, 27 (1983).

[27] Y. Nesterov, *Lectures on Convex Optimization*, Springer Publishing Company, Incorporated, 2nd ed., 2018.

[28] P. Ochs, *Unifying abstract inexact convergence theorems and block coordinate variable metric ipiano*, SIAM Journal on Optimization, 29 (2019), pp. 541–570, https://doi.org/10.1137/17M1124085.

[29] T. Pock and S. Sabach, *Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems*, SIAM Journal on Imaging Sciences, 9 (2016), pp. 1756–1787, https://doi.

org/10.1137/16M1064064.
[30] F. Pompili, N. Gillis, P.-A. Absil, and F. Glineur, *Two algorithms for orthogonal nonnegative matrix factorization with application to clustering*, Neurocomputing, 141 (2014), pp. 15–25.
[31] M. Razaviyayn, M. Hong, and Z. Luo, *A unified convergence analysis of block successive minimization methods for nonsmooth optimization*, SIAM Journal on Optimization, 23 (2013), pp. 1126–1153, https://doi.org/10.1137/120891009.
[32] M. Teboulle and Y. Vaisbourd, *Novel proximal gradient methods for nonnegative matrix factorization with sparsity constraints*, SIAM Journal on Imaging Sciences, 13 (2020), pp. 381–421, https://doi.org/10.1137/19M1271750.
[33] P. Tseng and S. Yun, *A coordinate gradient descent method for nonsmooth separable minimization*, Mathematical Programming, 117 (2009), pp. 387–423.
[34] Y. Xu and W. Yin, *A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion*, SIAM Journal on Imaging Sciences, 6 (2013), pp. 1758–1789, https://doi.org/10.1137/120887795, https://doi.org/10.1137/120887795.
[35] Y. Xu and W. Yin, *A fast patch-dictionary method for whole image recovery*, Inverse Problems & Imaging, 10 (2016), p. 563, https://doi.org/10.3934/ipi.2016012.
[36] Y. Xu and W. Yin, *A globally convergent algorithm for nonconvex optimization based on block coordinate update*, Journal of Scientific Computing, 72 (2017), pp. 700–734.
[37] S. Zhong and J. Ghosh, *Generative model-based document clustering: a comparative study*, Knowledge and Information Systems, 8 (2005), pp. 374–384.

SUPPLEMENTARY MATERIALS: Block Bregman Majorization Minimization with Extrapolation*

Le Thi Khanh Hien†, Duy Nhat Phan‡, Nicolas Gillis†, Masoud Ahookhosh§, and Panagiotis Patrinos¶

SM1. Facial features extracted by the ONMF algorithms on the Frey and Umist facial images. Figures SM1, and SM2 display the facial features extracted by BMM, BMME, A-BPALM and BIBPA for the Frey and Umist facial images, respectively. In Figure SM1, facial

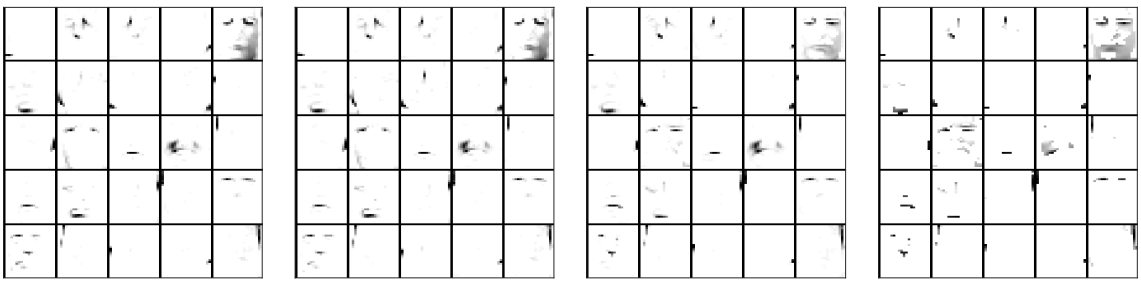

Figure SM1. *Display of the rows of the matrix V as facial features, computed by BMM, BMME, A-BPALM, and BIBPA on the Frey facial images.*

features are rather similar, although BMM and BMME obtained smaller objective function values.

In Figure SM2, as BMM and BMME both converged to similar objective function values (see Figure 2), they provide very similar facial features. A-BPALM and BIBPA were not able to converge within the 100 seconds, and hence provide worse facial features. For example, the first facial feature is much denser than for BMM and BMME, overlapping with other facial features (meaning that the orthogonality constraints is not well satisfied).

SM2. Scaled objective function values for document data sets. Figures SM3 and SM4 display the scaled objective function values for the document data sets on the penalized ONMF problem. We observe that, except for tr41 and tr45 where A-BPALM is able to compete with BMM and BMME, BMM and BMME outperform A-BPALM and BIBPA which performs particularly badly on these sparse data sets.

*LTK Hien and DN Phan contributed equally to this work.

Funding: The authors acknowledge the support by the European Research Council (ERC starting grant no 679515), and by the Fonds de la Recherche Scientifique - FNRS and the Fonds Wetenschappelijk Onderzoek - Vlaanderen (FWO) under EOS Project no O005318F-RG47.

†Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, Rue de Houdain 9, 7000 Mons, Belgium (thikhanhhien.le@umons.ac.be, nicolas.gillis@umons.ac.be).

‡Dynamic Decision Making Laboratory, Carnegie Mellon University, USA (dnphan@andrew.cmu.edu).

§Department of Mathematics, University of Antwerp, Belgium (masoud.ahookhosh@uantwerp.be).

¶Department of Electrical Engineering (ESAT-STADIUS), KU Leuven, Belgium (panos.patrinos@esat.kuleuven).

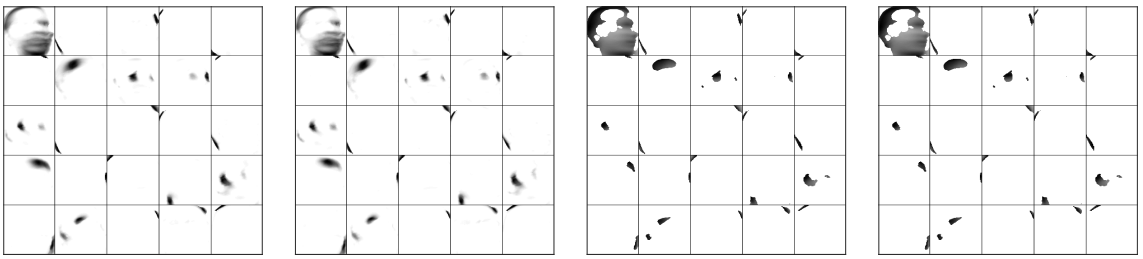

Figure SM2. *Display of the rows of the matrix V as facial features, computed by BMM, BMME, A-BPALM, and BIBPA on the Umist facial images.*

SM3. Comparison between BMME and CoCaIn on the matrix completion problem.

In this section, we consider BMME for solving the CSOP (1.1) with $m = 1$. As $m = 1$, we can omit the index i. In addition, the relative smooth parameters and the kernel generating distance do not depend on k. Therefore, the condition (3.1) can be rewritten as follows

$$D_\varphi(x^k, \bar{x}^k) \leq \frac{\delta L}{L + l} D_\varphi(x^{k-1}, x^k),$$

and the update (3.2) becomes

$$x^{k+1} \in \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ L D_\varphi(x, \bar{x}^k) + \left\langle \nabla f(\bar{x}^k), x \right\rangle + u(x, x^k) \right\}.$$

In some applications, the constants L, l might be very large, leading to a slow convergence. Hence, like CoCaIn [SM4] we incorporate a backtracking line search for L and l into BMME. In particular, BMME with backtracking computes the extrapolation point $\bar{x}^k = x^k + \beta^k(x^k - x^{k-1}) \in \operatorname{int} \operatorname{dom} \varphi$, where β^k satisfies the following condition

$$D_\varphi(x^k, \bar{x}^k) \leq \frac{\delta L^{k-1}}{L^{k-1} + l^k} D_\varphi(x^{k-1}, x^k),$$

where l^k is updated via backtracking such that

$$D_f(x^k, \bar{x}^k) \geq -l^k D_\varphi(x^k, \bar{x}^k).$$

The update x^{k+1} is computed by solving the following convex nonsmooth sub-problem

$$\min_{x \in \mathcal{X}} \left\{ L^k D_\varphi(x, \bar{x}^k) + \left\langle \nabla f(\bar{x}^k), x \right\rangle + u(x, x^k) \right\},$$

where L^k is chosen via backtracking such that $L^k \geq L^{k-1}$ and

$$D_f(x^{k+1}, \bar{x}^k) \leq L^k D_\varphi(x^{k+1}, \bar{x}^k).$$

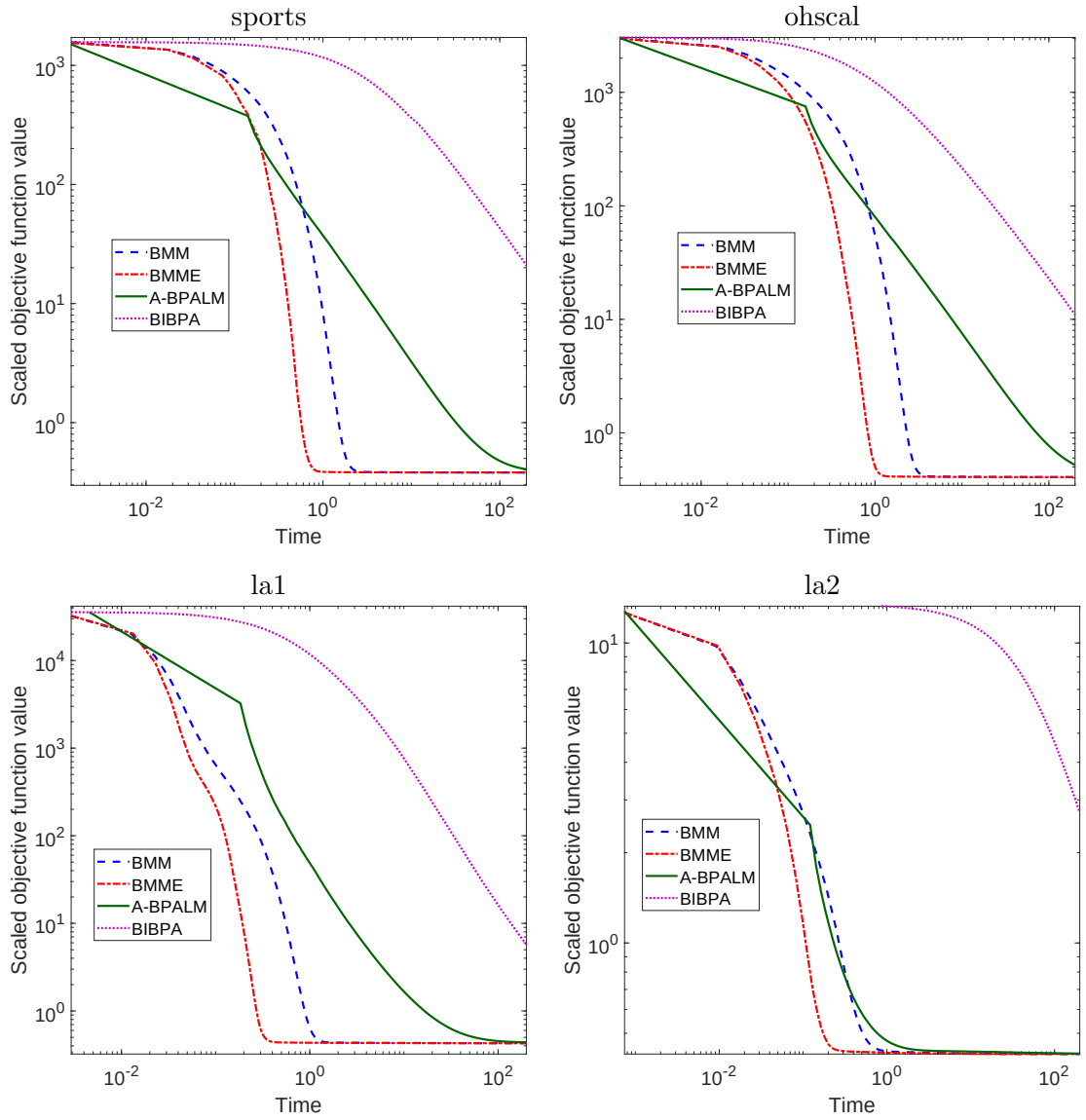

Figure SM3. *Evolution of scaled objective function values with respect to time on document data sets.*

We now conduct an additional experiment on the following matrix completion problem (MCP) to demonstrate the advantages of using proper convex surrogate functions:

$$\min_{U\in\mathbb{R}^{\mathbf{m}\times\mathbf{r}},V\in\mathbb{R}^{\mathbf{r}\times\mathbf{n}}}\left\{\frac{1}{2}\|\mathcal{P}(A-UV)\|_F^2+g(U,V)\right\}, \tag{SM3.1}$$

where $A\in\mathbb{R}^{\mathbf{m}\times\mathbf{n}}$ is a given data matrix, $\mathcal{P}(Z)_{ij}$ is equal to Z_{ij} if A_{ij} is observed and is equal to 0 otherwise, and g is a regularization term. Here, we are interested in an exponential

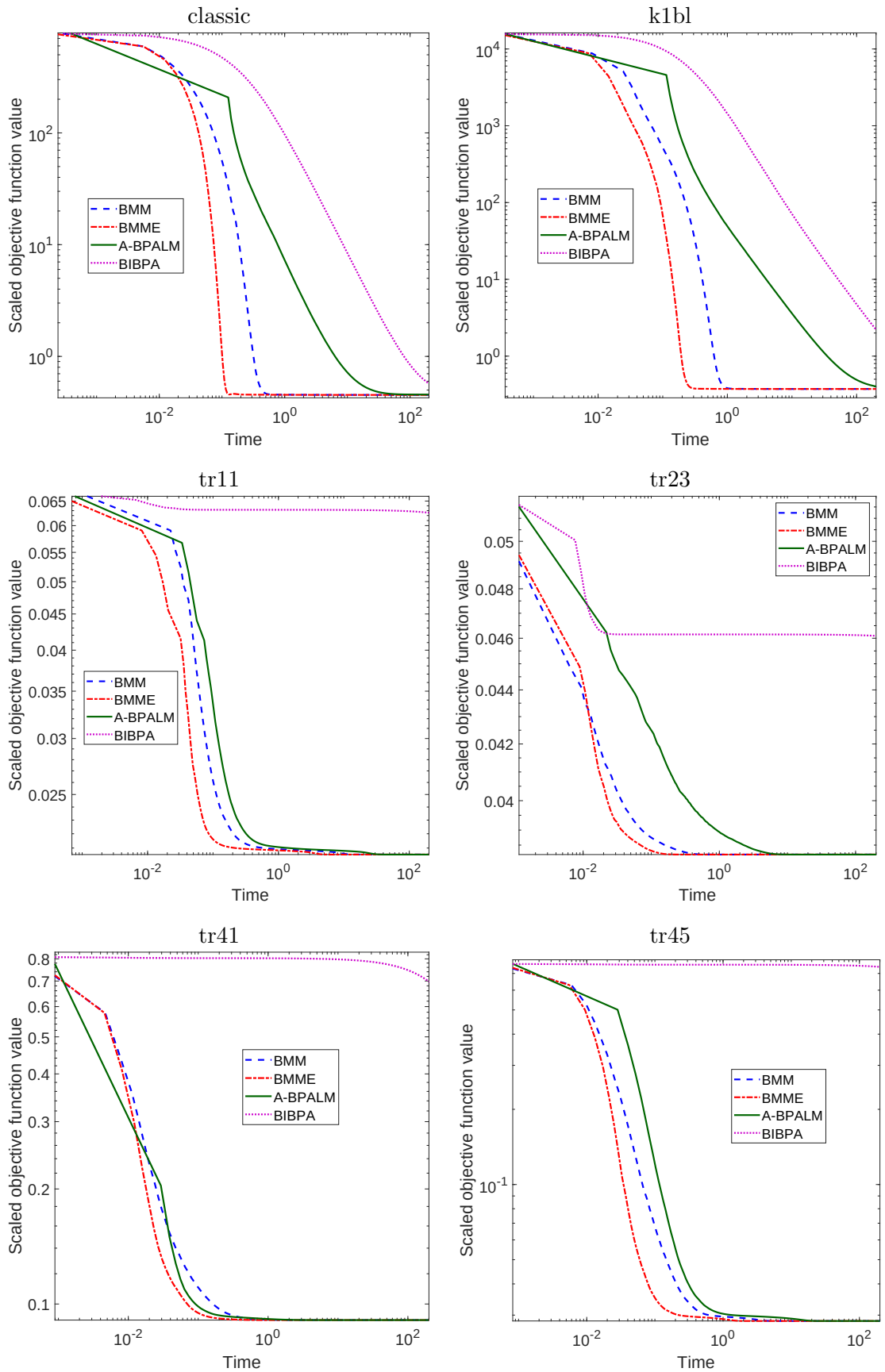

Figure SM4. *Evolution of scaled objective function values with respect to time on document data sets.*

regularization g defined by

$$g(U,V) = \lambda\Big(\sum_{ij}\big(1-\exp(-\theta|U_{ij}|)\big) + \sum_{ij}\big(1-\exp(-\theta|U_{ij}|)\big)\Big),$$

where λ and θ are tuning parameters. We consider the problem (SM3.1) as the form of (1.1) with $m=1$, $\mathcal{X}_1 = \mathcal{X} = \mathbb{R}^{\mathbf{m}\times\mathbf{r}} \times \mathbb{R}^{\mathbf{r}\times\mathbf{n}}$, $f(U,V) = \frac{1}{2}\|\mathcal{P}(A-UV)\|_F^2$, and $g_1(U,V) = g(U,V)$. We now investigate BMME for solving the problem (SM3.1) by choosing a kernel generating distance φ given by

$$\varphi(U,V) = c_1\left(\frac{\|U\|_F^2 + \|V\|_F^2}{2}\right)^2 + c_2\frac{\|U\|_F^2 + \|V\|_F^2}{2},$$

where $c_1 = 3$ and $c_2 = \|\mathcal{P}(A)\|_F$. In [SM3], the authors showed that f is (L,l)-relative smooth to φ for all $L, l \geq 1$. BMME iteratively chooses a convex surrogate function u of g as follows:

$$u(U,V,U^k,V^k) = g(U^k,V^k) + \langle W^{U^k}, |U|\rangle + \langle W^{V^k}, |V|\rangle,$$

where $W_{ij}^{U^k} = \lambda\theta\exp(-\theta|U_{ij}^k|)$. BMME with backtracking updates (U^{k+1}, V^{k+1}) by solving the following convex nonsmooth sub-problem

$$\min_{U,V}\Big\{u(U,V,U^k,V^k) + \langle P^k, U\rangle + \langle Q^k, V\rangle + L^k\varphi(U,V)\Big\}, \tag{SM3.2}$$

where $P^k = \nabla_U f(\bar{U}^k,\bar{V}^k) - L^k\nabla_U\varphi(\bar{U}^k,\bar{V}^k)$, $Q^k = \nabla_V f(\bar{U}^k,\bar{V}^k) - L^k\nabla_V\varphi(\bar{U}^k,\bar{V}^k)$, and L^k is chosen via a backtracking line search. The solution to the problem (SM3.2) is defined by $U^{k+1} = -\tau^*\mathcal{S}(P^k, W^{U^k})/L^k$ and $V^{k+1} = -\tau^*\mathcal{S}(Q^k, W^{V^k})/L^k$, where $\mathcal{S}(A,B)_{ij} = [|A_{ij}| - B_{ij}]_+\text{sign}(A_{ij})$, and τ^* is the unique positive real root of

$$c_1\left(\|\mathcal{S}(P^k, W^{U^k})/L^k\|_F^2 + \|\mathcal{S}(Q^k, W^{V^k})/L^k\|_F^2\right)\tau^3 + c_2\tau - 1 = 0.$$

Since CoCaIn [SM4] does not use the MM step and requires the weakly convexity of g, it is different from BMME for updating (U^{k+1}, V^{k+1}) and initializing L^0. In particular, CoCaIn iteratively solves the following nonconvex sub-problem

$$\min_{U,V}\Big\{g(U,V) + \langle P^k, U\rangle + \langle Q^k, V\rangle + L^k\varphi(U,V)\Big\},$$

which does not have closed-form solutions. We therefore employ an MM scheme to solve this sub-problem. For initializing the step-size, CoCaIn requires $L^0 > \frac{\lambda\theta^2}{(1-\delta-\epsilon)c_2}$ that might be quite large, where $\epsilon \in (0,1)$ such that $\delta+\epsilon < 1$. Unlike CoCaIn, our BMME with backtracking can use any L^0. The flexibility of the initialization L^0 may lead to a faster convergence.

In the experiment, we set $\lambda = 0.1$, $\theta = 5$, $\delta = 0.99$, and $l^0 = 0.001$. We initialize $L^0 = 1.0001\frac{\lambda\theta^2}{(1-\delta-\epsilon)c_2}$ with $\epsilon = 0.009$ for CoCaIn while we choose $L^0 = 0.01$ for BMME with backtracking. We carry out the experiment on MovieLens 1M that contains $999,714$ ratings

of 6,040 different users. We choose $\mathbf{r} = 5$ and randomly use 70% of the observed ratings for training and the rest for testing. The process is repeated twenty times. We run each algorithm 20 seconds. We are interested in the root mean squared error on the test set: $RMSE = \sqrt{\|\mathcal{P}_T(A - UV)\|^2/N_T}$, where $\mathcal{P}_T(Z)_{ij} = Z_{ij}$ if A_{ij} belongs to the test set and 0 otherwise, N_T is the number of ratings in the test set. We plotted the curves of the average value of RMSE and the objective function value versus training time in Figure SM5.

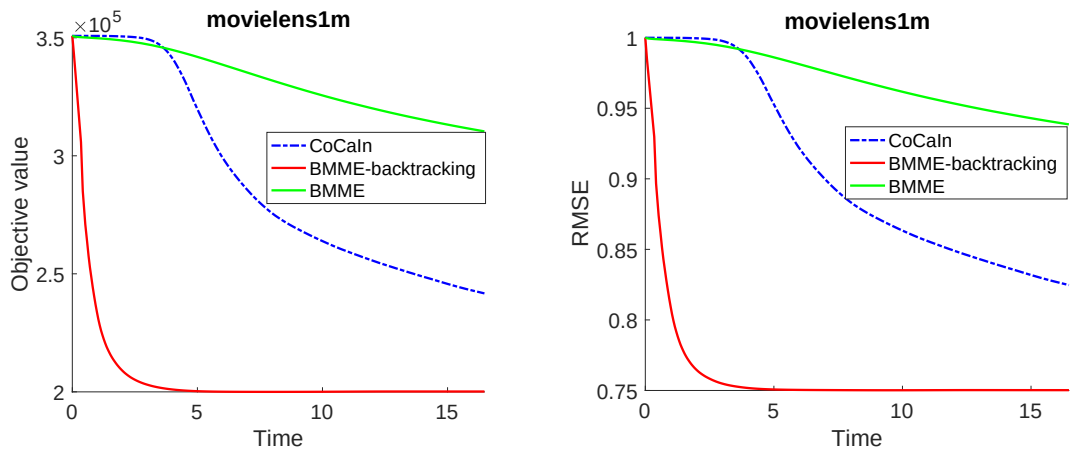

Figure SM5. *BMME and CoCaIn applied on the MCP (SM3.1). Evolution of the average value of the RMSE on the test set and the objective function value with respect to time.*

We observe that BMME with backtracking converges much faster than CoCaIn and BMME without backtracking (BMME). This illustrates the usefulness of properly choosing the convex surrogate function and the backtracking line search for the relative smooth constants.

SM4. Pseudo codes of A-BPALM and BIBPA. We provide the pseudo code of A-BPALM (that is, [SM2, Algorithm 2]) in Algorithm SM4.1 and the pseudo code of BIBPA (that is, [SM1, Algorithm 1]) in Algorithm SM4.2. Note that the kernel function h in Algorithm SM4.1 satisfies [SM2, Definition 3.4] and the kernel functions $h_1, \ldots, h_m$ in Algorithm SM4.2 satisfy [SM1, Definition 2.2].

REFERENCES

[1] M. Ahookhosh, L. T. K. Hien, N. Gillis, and P. Patrinos, *A block inertial Bregman proximal algorithm for nonsmooth nonconvex problems with application to symmetric nonnegative matrix tri-factorization*, Journal of Optimization Theory and Applications, (2021).

[2] M. Ahookhosh, L. T. K. Hien, N. Gillis, and P. Patrinos, *Multi-block Bregman proximal alternating linearized minimization and its application to sparse orthogonal nonnegative matrix factorization*, Computational Optimization and Application, 79 (2021), p. 681715.

[3] M. C. Mukkamala and P. Ochs, *Beyond alternating updates for matrix factorization with inertial Bregman proximal gradient algorithms*, in Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, eds., 2019, pp. 4268–4278.

Algorithm SM4.1 A-BPALM

Require: $x^0 \in \operatorname{int}\mathbf{dom}h$, $\nu > 1$, $\overline{L}_i^0 > 0$ for $i = 1, \ldots, m$, $I_n = (U_1, \ldots, U_m) \in \mathbb{R}^{n\times n}$ with $U_i \in \mathbb{R}^{n\times n_i}$ and the identity matrix I_n.
Let $k = 0$, $p = 0$, $\gamma_i^0 \in \left(0, \frac{1}{\overline{L}_i^0}\right)$ for $i = 1, \ldots, m$.
while some stopping criterion is not met **do**
  $x^{k,0} = x^k$;
  for $i = 1, \ldots, m$ **do**
    repeat
      set $\overline{L}_i^{k+1} = \nu^p \overline{L}_i^k$, $\gamma_i^{k+1} = \frac{\gamma_i^k}{\nu^p}$, $p = p + 1$ and compute

$$x_i^{k,i} \in \operatorname*{argmin}_{z_i \in \mathbb{R}^{n_i}} f(x^{k,i-1}) + \left\langle \nabla f(x^{k,i-1}), U_i(z_i - x_i)\right\rangle + \frac{1}{\gamma_i^{k+1}} D_h\left(x^{k,i-1} + U_i(z_i - x_i^k), x^{k,i-1}\right) + g_i(z_i)$$

$$x^{k,i} = x^{k,i-1} + U_i(x_i^{k,i} - x_i^{k,i-1});$$

    until $f(x^{k,i}) \le f(x^{k,i-1}) + \left\langle \nabla_i f x^{k,i-1}), x_i^{k,i} - x_i^{k,i-1}\right\rangle + \overline{L}_i^{k+1} D_h(x^{k,i}, x^{k,i-1})$
    $p_i^k = p - 1$; $p = 0$;
  end for
  $x^{k+1} = x^{k,m}$, $k = k + 1$;
end while
Ensure: a vector x^k.

Algorithm SM4.2 BIBPA

Require: $x^0 \in \operatorname{int}\mathbf{dom}\, h_1$, $I_n = (U_1, \ldots, U_m) \in \mathbb{R}^{n\times n}$ with $U_i \in \mathbb{R}^{n\times n_i}$ and the identity matrix I_n, $k = 0$.
while some stopping criterion is not met **do**
  $x^{k,0} = x^k$;
  for $i = 1, \ldots, m$ **do**
    choose γ_i^k and α_i^k satisfying [SM1, Proposition 3.5] and compute

$$x_i^{k,i} \in \operatorname*{argmin}_{z_i \in \mathbb{R}^{n_i}} \left\langle \nabla_i f(x^{k,i-1}) - \frac{\alpha_i^k}{\gamma_i^k}(x_i^k - x_i^{k-1}), z_i - x_i^k\right\rangle + \frac{1}{\gamma_i^k} D(x^{k,i-1} + U_i(z_i - x_i^k), x^{k,i-1}) + g_i(z_i);$$

$$x^{k,i} = x^{k,i-1} + U_i(x_i^{k,i} - x_i^{k,i-1});$$

  end for
  $x^{k+1} = x^{k,m}$, $k = k + 1$;
end while
Ensure: a vector x^k.

[4] M. C. Mukkamala, P. Ochs, T. Pock, and S. Sabach, *Convex-concave backtracking for inertial Bregman proximal gradient algorithms in nonconvex optimization*, SIAM Journal on Mathematics of Data Science, 2 (2020), pp. 658–682, https://doi.org/10.1137/19M1298007.