symmetry

MDPI

Article

A Gradient-Based Algorithm with Nonmonotone Line Search for Nonnegative Matrix Factorization

Wenbo Li [1,*] and Xiaolu Shi [2]

[1] School of Sciences, Xi'an University of Technology, Xi'an 710054, China
[2] Beijing Mechanical Equipment Institute, Beijing 100039, China; shixiaolu@ht206.casic.com.cn
* Correspondence: wbli57@hotmail.com or 105293@xaut.edu.cn

Abstract: In this paper, we first develop an active set identification technique, and then we suggest a modified nonmonotone line search rule, in which a new parameter formula is introduced to control the degree of the nonmonotonicity of line search. By using the modified line search and the active set identification technique, we propose a global convergent method to solve the NMF based on the alternating nonnegative least squares framework. In addition, the larger step size technique is exploited to accelerate convergence. Finally, a large number of numerical experiments are carried out on synthetic and image datasets, and the results show that our presented method is effective in calculating speed and solution quality.

Keywords: active set; alternating nonnegative least squares; projected barzilai-borwein method; nonmonotone line search; larger step size

Citation: Li, W.; Shi, X. A Gradient-Based Algorithm with Nonmonotone Line Search for Nonnegative Matrix Factorization. *Symmetry* **2024**, *16*, 154. https://doi.org/10.3390/sym16020154

Academic Editors: Jianchao Bai, Qiyu Jin, Yuchao Tang and Marco Montemurro

Received: 2 November 2023
Revised: 14 December 2023
Accepted: 27 December 2023
Published: 29 January 2024

1. Introduction

As a typical nonnegative data dimensionality reduction technology, nonnegative matrix factorization (NMF) [1–5] can efficiently mine hidden information from data, so it has been gradually applied to research into high-dimensional data. This method as a data reduction technique appears in many applications, such as image processing [2], text mining [6], blind source separation [7], clustering [8], music analysis [9], and hyperspectral imaging unmixing [10], to name a few. Generally speaking, the fundamental NMF problem can be summarized as follows: given an $m \times n$ data matrix $V = (V_{ij})$ with $V_{ij} \geq 0$ and a predetermined positive integer $r < \min(m, n)$, then NMF plans to find two nonnegative matrices $W \in \mathbb{R}_+^{m\times r}$ and $H \in \mathbb{R}_+^{r\times n}$ such that

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

Our visualization illustration of NMF is shown in Figure 1.

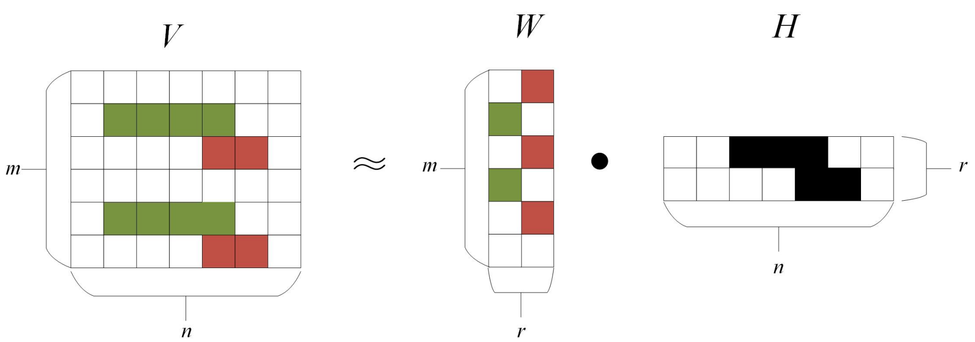


Figure 1. Visualization illustration of NMF.

One of the most commonly used models of NMF (1) is

$$\begin{aligned} &\min_{W,H} f(W,H) \equiv \frac{1}{2}\|V - WH\|_F^2 \\ &\text{subject to} \quad W \geq 0, H \geq 0. \end{aligned} \tag{2}$$

where $\|\cdot\|_F$ is the Frobenius norm.

The project Barzilai-Borwein (PBB) algorithm is regarded as a popular and effective method for solving (2) which was originated by Barzilai and Borwein [11]. In recent years, a large number of studies [12–16] have shown that the PBB algorithm is a very effective algorithm in solving optimal problems. The PBB algorithm has the characteristics of simple calculation and high efficiency, so it has been paid attention to by various disciplines. So far, the research results based on the PBB have been widely used in the field of NMF (see [17–21]).

In view of the perfect symmetry of the interaction between W and H, we will focus on the updating of matrix W based on the PBB algorithm. Remember that H^k is an approximate value of H after kth update, and there are

$$f(W, H^k) = \frac{1}{2}\|V - WH^k\|_F^2 \quad \forall k. \tag{3}$$

At each step for solving (3), there are three different updates:

$$W^{k+1} = \min_{W \geq 0} f(W, H^k); \tag{4}$$

$$W^{k+1} = \min_{W \geq 0} f(W, H^k) + \langle \nabla f(W), W - W^k \rangle + \frac{L_W^k}{2}\|W - W^k\|_F^2; \tag{5}$$

$$W^{k+1} = \min_{W \geq 0} \langle \nabla f(W^k), W - W^k \rangle + \frac{L_W^k}{2}\|W - W^k\|_F^2, \tag{6}$$

where $L_W^k > 0$, $\nabla f(W) = \nabla_W f(W, H^k)$.

Original cost function (4) is the most frequently used form in the PBB method for NMF and has been widely and deeply researched [17,20–23]. But the major disadvantage of (4) is that it is not strongly convex [24–28], and we can only hope that this method can find a stationary point, rather than a global or local minimizer. To overcome this drawback, a proximal modification of cost function (4) is presented in [18,19], namely, the proximal cost function (5).

At present, the proximal cost function (5) has been used with the PBB method for NMF in [18,19]. When the cost function (5) is a strongly convex quadratic optimization problem, their lower bound is zero, so the subproblem (5) has a unique minimizer. In [18], the authors present a quadratic regularization nonmonotone PBB algorithm to solve (5) and established its global convergence result under mild conditions. Recently, it is revisited in [19] for the monotone PBB method and is also shown to converge globally to a stationary point of (3), and through the analysis of numerical experiments, it is proved that the monotone PBB method can win over the nonmonotone one under certain conditions. However, when solving the problems (4) and (5), the existing gradient methods based on the PBB converge slowly due to the nonnegative conditions. Therefore, this project intends to develop a new fast NMF algorithm.

In this paper, we introduce a prox-linear approximation of $f(W, H^k)$ at W^k based on $\nabla f(W)$ which is the cost function (6). And then we propose an active set identification technique. Next, we present a modified nonmonotone line search technique so as to improve the efficiency of nonmonotone line search, in which a new parameter formula is presented to attempt to control the degree of the nonmonotonicity of line search, and thus improve both the possibility of finding the global optimal solution and the convergence speed. By using the active set identification strategy and the modified nonmonotone line search, a global convergent method is proposed to solve (6) based on the alternating nonnegative least squares framework. In particular, in each iteration, identification techniques are used to determine active and free variables. We take $(D_t)_{ij} = 0$ or $(D_t)_{ij} = -(Z_t)_{ij}$ to update some active variables, while using a projected Barzilai-Borwein method to update the free variables and some active variables. The calculation speed is improved by using the method of larger step size. Finally, through the numerical experiments of simulation data and image data, it is proved that the proposed algorithm is effective.

This paper is organized in the following manner. In Section 3, we introduce our estimation of active set, put forward an efficient NMF algorithm, and present the global convergence results of this method. The experimental results are given in Section 4. Finally, Section 5 is the conclusion of the thesis.

2. A Fast PBB Algorithm

In this section, we present an efficient algorithm for solving the NMF and establish the global convergence of our algorithm. Now, let us first introduce some main results of the objective function $f(W, H^k)$ that we know.

Lemma 1 ([29]). *The following two statements are valid.*

(i) *The objective function $f(W, H^k)$ of (3) is convex.*
(ii) *The gradient*

$$\nabla_W f(W, H^k) = (WH^k - V)(H^k)^T$$

is Lipschitz continuous with the constant $L_W = \|H^k(H^k)^T\|_2$.

In order to facilitate the discussion, we mainly focus on (6) and then rewrite it. Note that the cost function (7) is closely related to the one in Xu et al. [30], but has the following difference: matrix U is W_t in our cost function (7), however, to [30] the matrix U is an extrapolation point in W_t.

$$\min_{W \geq 0} \varphi(U, W) := \langle \nabla f(U), W - U \rangle + \frac{L_W}{2} \|W - U\|_F^2, \tag{7}$$

where the fixed matrix $U \geq 0$.

According to (ii) of the Lemma 1, $\varphi(U, W)$ is strictly convex in W for any given U. In each iteration, we will first solve the following strongly convex quadratic minimization problem, so as to obtain a Z_t value

$$\min_{W \geq 0} \varphi(W_t, W). \tag{8}$$

Because the objective function of the problem (8) is strongly convex, the solution of the problem is unique and closed-form

$$Z_t = P[W_t - \frac{1}{L_W} \nabla_W f(W_t, H^k)], \tag{9}$$

Here, the operator $P[X]$ projects all negative terms of X to zero.

Let $W_{t+1} = Z_t + D_t$, where D_t is the direction which is obtained by (23) with α_t being the BB stepsize [11], whereby we see that the convergence of $\{W_{t+1}\}$ can not be guaranteed. Therefore, a global optimization strategy is proposed based on the modified Armiji line search [31].

Therefore, a globalization strategy based on the modified Armiji line search [31] has been proposed, that is, we ask for a step size λ_t, so that

$$f(Z_t + \lambda_t D_t) \leq \max_{0 \leq j \leq min\{t, M-1\}} f(Z_{t-j}) + \gamma \lambda_t \langle \nabla f(Z_t), D_t \rangle, \tag{10}$$

here $M > 0$. Owing to the maximum function, a good function value obtained in any iteration will be discarded, and the numerical performance depends largely on the selection of M in some cases (see [32]).

So as to overcome these shortcomings and obtain a large step size in each procedure, we present a modified nonmonotone line search rule. The modified line search is as follows: for the known iteration point Z_t and search direction D_t at Z_t, we select $\eta_t \in [\eta_{min}, \eta_{max}]$, where $0 < \eta_{min} < \eta_{max} < 1$, $\gamma_t \in [\gamma_{min}, \gamma_{max}(1-\eta_{max})]$, where $\gamma_{max} < 1$, $0 < \gamma_{min} < (1-\eta_{max})\gamma_{max}$, $0 \leq \mu \leq 1$, and $s > 1$, to find a λ_t satisfying the following inequality:

$$S_{t+1} \leq S_t + \gamma_t \lambda_t [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2], \tag{11}$$

where S_t is defined as

$$S_t = \begin{cases} f(W_0), & \text{if} \quad t = 0, \\ f(W_t) + \eta_{t-1}(S_{t-1} - f(W_t)), & \text{if} \quad t \geq 1, \end{cases} \tag{12}$$

Similar to M in (10), the selection η_t in (12) is an important factor in determining the degree of nonmonotonicity (see [33]). Thus, to improve the efficiency of a nonmonotone line search, Ahookhosh et al. [34] choose a varying value for the parameter η_t by using a simple formula. Later, Nosratipour et al. [35] decided that η_t should be related to a suitable criterion to measure the distance to the optimal solution. Thus, they defined η_t by

$$\eta_t = 1 - e^{-\|\nabla f(Z_t)\|}. \tag{13}$$

However, we found that if the iterative sequence $\{Z_t\}$ is trapped in a narrow curved valley, then it can lead to $\nabla f(Z_t) = 0$, from which we can obtain $\eta_t = 0$, so the nonmonotone line search is reduced to the standard Armijo line search, which is inefficient owing to the generation of very short or zigzagging steps. To overcome this drawback, we suggest the following η_t:

$$\eta_t = \frac{2}{\pi} arctan(|f(Z_t) - f(Z_{t-1})|). \tag{14}$$

It is obvious that $|f(Z_t) - f(Z_{t-1})|$ is large when the function value decreases rapidly, and then η_t will also be large, so therefore the nonmonotone strategy will be stronger. However, when $f(Z_t)$ is close to the optimal solution, we can obtain $|f(Z_t) - f(Z_{t-1})|$ which tends toward zero, and then η_t also tends toward zero, so then the nonmonotone rule will be weaker and it tends to be a monotone rule.

As was observed in [16], the active set method can enhance the efficiency of the local convergence algorithm and reduce the computing cost. There-in-after, we will recommend an active set recognition technology to approximate the right sustain of the solution points. In our context, we deal with the active set which is considered as the subset of zero components of Z^*. Now, we introduce the active set L as the index set corresponding to the zero component. Meanwhile, the inactive set F is to be the support of Z^*.

Definition 1. *Let $\Omega = \{Z \in \mathbb{R}^{m\times r} : Z \geq 0\}$ and Z^* be a stationary point of (3). We define the active set as follows:*

$$L = \{ij : Z^*_{ij} = 0\}, \tag{15}$$

We further define an inactive set F which is a complementary set of L,

$$F(Z) = I \backslash L(Z), \tag{16}$$

where $I = \{11, 12, \ldots, 1r, 21, 22, \ldots, 2r, \ldots, m1, m2, \ldots, mr\}$.

Then, for any $(Z_t) \in \Omega$, we define the following approximations $L(Z_t)$ and $F(Z_t)$ as $\bar{L}$ and $\bar{F}$, respectively,

$$L(Z_t) = \{ij : (Z_t)_{ij} \leq \frac{1}{\alpha_t}\nabla f(Z_t)_{ij}\}, \tag{17}$$

$$F(Z_t) = I \backslash L(Z_t), \tag{18}$$

where α_t is the BB step size. For simplicity, we abbreviate $L(Z_t)$ and $F(Z_t)$ as L_t and F_t, respectively. Similar to the Lemma 1 in [21], we have that if the strict complementarity is satisfied at Z_t, then $L(Z_t)$ coincides with the active set if Z_t is sufficiently close to Z^*.

In order to obtain a well estimate of the active set, the active set is further subdivided into two sets

$$L_1(Z_t) = \{ij \in L(Z_t) : \nabla f(Z_t)_{ij} \geq c\}, \tag{19}$$

and

$$L_2(Z_t) = \{ij \in L(Z_t) : \nabla f(Z_t)_{ij} < c\}, \tag{20}$$

here $c > 0$ is a constant.

Obviously, $L_2(Z_t)$ is the index set of variables with the first-order necessary condition. Therefore, we have reason to set the variables with indices in $L_2(Z_t)$ to 0. In addition, because $L_1(Z_t)$ is an index set that does not satisfy the first-order necessary condition, we further subdivide $L_1(Z_t)$ into two subsets

$$\bar{L}_1(Z_t) = \{ij : ij \in L_1(Z_t) \text{ and } (Z_t)_{ij} = 0\}, \tag{21}$$

and

$$\tilde{L}_1(Z_t) = \{ij : ij \in L_1(Z_t) \text{ and } (Z_t)_{ij} \neq 0\}. \tag{22}$$

When a variable is with indices in $\bar{L}_1(Z_t)$, we consider the direction of the form 0. And for the variables of the indexs in $\tilde{L}_1(Z_t)$, we consider the direction of the form $-Z_t$, so as to to improve the corresponding components. Thus, through the above discussion, we define this direction in the following compact form:

$$(D_t)_{ij} = \begin{cases} 0, & \text{if } ij \in \bar{L}_1(Z_t), \\ -(Z_t)_{ij}, & \text{if } ij \in \tilde{L}_1(Z_t), \\ (P[Z_t - \alpha_t \nabla f(Z_t)] - Z_t)_{ij}, & \text{if } ij \in L_2(Z_t) \cup F(Z_t), \end{cases} \tag{23}$$

where α_t is the BB stepsize.

Finally, we let

$$W_{t+1} = Z_t + \lambda_t D_t, \tag{24}$$

where λ_t is the step size which is found by using a nonmonotonic line search (11).

It is known from [36] that the larger step size technique can significantly accelerate the rate of convergence of the algorithm, so by adding a relaxation factor s to the update rule of W_{t+1} (24), we modify the update rule (24) as

$$W_{t+1} = Z_t + s\lambda_t D_t \tag{25}$$

for relaxation factor $s > 1$. We show that the optimal parameter s in (25) is $s = 1.7$ by number experiments in Section 4.4.

Based on the above discussion, we develop a nonmonotone projected Barzilai-Borwein method based on the active set strategy proposed in Section 3 and outline the proposed algorithm in Algorithm 1. We can follow a similar procedure for updating H.

Algorithm 1 Nonmonotone projected Barzilai-Borwein algorithm (NMPBB).

1. Initialize $\alpha_0 = 1$, $\eta_t \in (0,1)$, choose parameters $\eta_t \in [\eta_{min}, \eta_{max}]$, $\gamma_t \in [\gamma_{min}, \gamma_{max}(1-\eta_{max})]$, $\alpha_{max} > \alpha_{min} > 0$, $\mu \in [0,1], \rho \in (0,1)$, $s > 1$, $L_W = \|H^k(H^k)^T\|_2$ and $W_0 = W^k$. Set $t = 0$.
2. If $\|P[W_t - \nabla f(W_t)] - W_t\| = 0$, stop.
3. Compute $Z_t = P[W_t - \frac{1}{L_W}\nabla f(W_t, H^k)]$.
4. Compute S_t by (12) and compute D_t by (23).
5. Nonmonotone line search. Let m_t be the smallest nonnegative integer m satisfying
$$S_{t+1} \leq S_t + \gamma_t \rho^m [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t}\|D_t\|^2], \tag{26}$$
where $D_t = P[Z_t - \alpha_t \nabla f(Z_t)] - Z_t$. Set $\lambda_t = \rho^{m_t}$, calculate $W_{t+1} = Z_t + s\lambda_t D_t$.
6. Calculate $X_t = W_{t+1} - Z_t$ and $Y_t = \nabla f(W_{t+1}) - \nabla f(Z_t)$. If $\langle X_t, X_t \rangle / \langle X_t, Y_t \rangle \leq 0$, set $\alpha_{t+1} = \alpha_{max}$; otherwise, set $\alpha_{t+1} = \min\{\alpha_{max}, \max\{\alpha_{min}, \langle X_t, X_t \rangle / \langle X_t, Y_t \rangle\}\}$.
7. Set $t = t+1$ and go to step 2.

Remark 1. *According to (11), from the definition of S_t, we obtain*
$$(1-\eta_t) f(Z_t + s\lambda_t D_t) \leq (1-\eta_t) S_t + \gamma_t \lambda_t [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t}\|D_t\|^2].$$
Since $\eta_t < 1$, we can find that (11) equals
$$f(Z_t + s\lambda_t D_t) \leq S_t + \frac{1}{1-\eta_t}\gamma_t \lambda_t [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t}\|D_t\|^2]. \tag{27}$$
If γ_{min} and γ_{max} are close to 0 and 1, respectively, and $\mu = 0$, then (11) reduces to the Gu's line search in [33] with $\gamma_t = \frac{\gamma}{1-\eta_t}$ and $\gamma \in [\gamma_{min}(1-\eta_t), \gamma_{max}]$, which implies that the linear search condition of Gu in [33] can be regarded as a special case of (11). In addition, when $\mu = 0$ and $\eta_t = 0$, the line search rule (11) can be reduced to the Armijo line search rule.

Next, we prove that the improved nonmonotone line search is well-defined. Before presenting this fact, we state the scaled projected gradient direction by
$$D_\alpha(W) = P[W - \alpha \nabla f(W)] - W \tag{28}$$
for all $\alpha > 0$ and $W \geq 0$.

For each $\alpha > 0$ and $W \geq 0$. The next Lemma 2 is very important in our proof.

Lemma 2 ([37])**.** *For each $\alpha \in (0, \alpha_{max}]$, $W \geq 0$,*

(i) $\langle \nabla f(W), D_\alpha(W) \rangle \leq -\frac{1}{\alpha}\|D_\alpha(W)\|^2 \leq -\frac{1}{\alpha_{max}}\|D_\alpha(W)\|^2$,
(ii) *The stationary point of (3) is at W if and only if $D_\alpha(W) = 0$.*

The lemma that follows states that $D_t = 0$ is true if and only if the stationary point of problem (3) is the iteration point $\{Z_t\}$.

Lemma 3. *Let D_t be calculated by (23), then $D_t = 0$ if and only if Z_t is a stationary point of problem (3).*

Proof. Let $(D_t)_{ij} = 0$. It is obvious that $(Z_t)_{ij}$ is a stationary point of problem (3) when $ij \in \bar{L}_1(Z_t)$. If $ij \in \tilde{L}_1(Z_t)$, we have
$$0 = (D_t)_{ij} = -(Z_t)_{ij} \geq -\frac{1}{\alpha_t}\nabla f(Z_t)_{ij}.$$

The above inequality implies that $\nabla f(Z_t)_{ij} \geq 0$. By the KKT condition, we can find that $(Z_t)_{ij}$ is a stationary point of problem (3). If $(D_t)_{ij} = 0, ij \in L_2(W_t) \cup F(W_t)$, by (ii) of Lemma 2, we know that $(Z_t)_{ij}$ is a stationary point of problem (3).

Assume that Z_t is a stationary point of (3). From the KKT condition, (17) and (18), we have

$$\bar{L}_t = \{ij : (Z_t)_{ij} = 0\},\ \bar{F}_t = \{ij : (Z_t)_{ij} > 0\}.$$

By the definition of $(D_t)_{ij}$, we have $(D_t)_{ij} = 0$ for all $ij \in L_1(Z_t)$. And then from the (ii) of Lemma 2, we have $(D_t)_{ij} = 0$ for all $ij \in L_2(Z_t)$. Therefore, we have $(D_t)_{ij} = 0$ for all $ij \in \bar{L}(Z_t)$. For another case, since $\nabla f(Z_t)_{ij} = 0$, for $ij \in \bar{F}_t$, and $\{Z_t\}_{ij}$ is a feasible point, from the definition of $(D_t)_{ij}$, we have $(D_t)_{ij} = 0, \forall ij \in \bar{F}_t$. □

The next Lemma 4 is very important in our proof.

Lemma 4. *Sequence $\{Z_t\}$ produced by Algorithm 1, we have*

$$\langle \nabla f(Z_t), D_t(Z_t) \rangle \leq -\frac{1}{\alpha_t} \|D_t(Z_t)\|^2, \tag{29}$$

$$\|D_t(Z_t)\| \leq \alpha_t \|\nabla f(Z_t)\|. \tag{30}$$

Proof. By (23), we know

$$D_{ij} = \begin{cases} 0, & \text{if } ij \in \bar{L}_1(Z_t), \\ -(Z_t)_{ij}, & \text{if } ij \in \tilde{L}_1(Z_t), \\ (P[Z_t - \alpha_t \nabla f(Z_t)] - Z_t)_{ij}, & \text{if } ij \in L_2(Z_t) \cup F(Z_t). \end{cases}$$

If $ij \in \bar{L}_1(Z_t)$, it is obvious that $\langle \nabla f(Z_t)_{ij}, (D_t(Z_t))_{ij} \rangle \leq -\frac{1}{\alpha_t} \|(D_t(Z_t))_{ij}\|^2$ holds. If $ij \in L_2(Z_t) \cup F(Z_t)$, from (i) of Lemma 2, we have

$$\langle \nabla f(Z_t)_{ij}, (D_t(Z_t))_{ij} \rangle \leq -\frac{1}{\alpha_t} \|(D_t(Z_t))_{ij}\|^2. \tag{31}$$

Thus, we now only need to prove that

$$\langle \nabla f(Z_t)_{ij}, (D_t(Z_t))_{ij} \rangle \leq -\frac{1}{\alpha_t} \|(D_t(Z_t))_{ij}\|^2, \quad \forall ij \in \tilde{L}_1(Z_t). \tag{32}$$

If $(D_t(Z_t))_{ij} = 0$, the inequality (32) holds. If $(D_t(Z_t))_{ij} \neq 0$, for all $ij \in \tilde{L}_1(Z_t)$, from (21), we have

$$(D_t(Z_t))_{ij} = -(Z_t)_{ij} \quad \text{and} \quad (Z_t)_{ij} \leq \frac{1}{\alpha_t} \nabla f(Z_t)_{ij},$$

which lead to

$$\langle \nabla f(Z_t)_{ij}, (D_t(Z_t))_{ij} \rangle \leq -\frac{1}{\alpha_t} \|(D_t(Z_t))_{ij}\|^2, \quad \forall ij \in \tilde{L}_1(Z_t). \tag{33}$$

The above deduction implies that the inequality (29) holds for $ij \in \bar{L}_1(Z_t)$. Combining (13) and (33), we obtain that (29) holds. By means of the Cauchy equality, from (29), we obtain (30). □

The following lemma is borrowed from Lemma 3 [18].

Lemma 5 ([18]). *Suppose Algorithm 1 generates $\{Z_t\}$ and $\{W_t\}$, there is*

$$f(Z_t) \leq f(W_t) - \frac{L_W}{2} \|Z_t - W_t\|^2 \tag{34}$$

Now, we will show the nice property of our line search.

Lemma 6. *Suppose Algorithm 1 generates sequences* $\{Z_t\}$ *and* $\{W_t\}$*, there is*

$$f(W_t) \leq S_t. \tag{35}$$

Proof. Based on the definition of S_t, we have

$$\begin{aligned} S_t - S_{t-1} &= f(W_t) + \eta_{t-1}(S_{t-1} - f(W_t)) - S_{t-1} \\ &= (1 - \eta_{t-1})(f(W_t) - S_{t-1}) \leq 0, \end{aligned} \tag{36}$$

where the last inequality from Lemma 2 and $\mu \in [0,1]$. From $1 - \eta_{t-1} > 0$, it concludes that $f(W_t) - S_{t-1} \leq 0$, i.e., $f(W_t) \leq S_{t-1}$.

Therefore, if $\eta_{t-1} \neq 0$, from (12), we have

$$\begin{aligned} S_t - f(W_t) &= f(W_t) + \eta_{t-1}(S_{t-1} - f(W_t)) - f(W_t) \\ &= \eta_{t-1}(S_{t-1} - f(W_t)) \\ &\geq 0 \end{aligned} \tag{37}$$

where the last inequality follows from (36). Thus, (37) indicates

$$f(W_t) \leq S_t. \tag{38}$$

In addition, if $\eta_{t-1} = 0$, we have $f(W_t) = S_t$. □

It follows from Lemma 6 that

$$f(W_t) \leq S_t \leq S_0 = f(W_0).$$

In addition, for any initial iterate $W_0 \geq 0$, Algorithm 1 generates sequences $\{Z_t\}$ and $\{W_t\}$ that are both included in the level set.

$$\mathcal{L}(W_0) = \{W | f(W) \leq f(W_0), W \geq 0\}.$$

Again, from Lemma 6, the theorem shown below can be easily obtained.

Theorem 1. *Assume that the level set* $\mathcal{L}(W_0)$ *is bounded, so the sequence* $\{S_t\}$ *is convergent.*

Proof. First, we show that $\{W_t\} \subset \mathcal{L}(W_0)$. Apparently, according to (35) we have

$$f(W_t) \leq S_t \leq S_{t-1} \leq \ldots \leq S_0 = f(W_0) \quad \forall t \in \mathbf{N}. \tag{39}$$

Therefore, we obtain that $\{W_t\} \subset L(W_0)$ for all $t \in \mathbf{N}$.

From (39), we can obtain that

$$\exists \tau \geq 0 \; s.t. \; \forall n \in \mathbf{N} : \tau \leq f(W_{t+n}) \leq S_{t+n} \leq S_{t-1+n} \leq \ldots \leq S_{t+1} \leq S_t,$$

that is, the sequence $\{S_t\}$ has a lower bound. Since the sequence $\{S_t\}$ is nonincreasing, the sequence $\{S_t\}$ is convergent. □

Next, we will exhibit that the line search (11) is well-defined.

Theorem 2. *Assume Algorithm 1 generates sequences* $\{Z_t\}$ *and* $\{W_t\}$*, so step 5 of the Algorithm 1 is well-defined.*

Proof. For this purpose, we prove that the line search stops at a limited value of steps. To establish a contradiction, we suppose that λ_t such that (26) does not exist, and then for all adequately large positive integers m, according to Lemmas 5 and 6, we have

$$S_{t+1} > S_t + \gamma_t \rho^m [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2], \tag{40}$$

According to (40), from the definition of S_t, we have

$$(1 - \eta_t) f(Z_t + s\rho^m D_t) > (1 - \eta_t) S_t + \gamma_t \rho^m [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2].$$

Since $\eta_t < 1$, we can find that (40) is equivalent to

$$f(Z_t + s\rho^m D_t) > S_t + \frac{1}{1 - \eta_t} \gamma_t \rho^m [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2]. \tag{41}$$

From Lemmas 5 and 6, we have

$$f(Z_t + s\rho^m D_t) > f(Z_t) + \frac{\gamma_t \rho^m}{(1 - \eta_t)} [\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2].$$

Due to $\langle \nabla f(Z_t), D_t \rangle + \frac{\mu}{\alpha_t} \|D_t\|^2 \geq \langle \nabla f(Z_t), D_t \rangle$, thus,

$$f(Z_t + s\rho^m D_t) - f(Z_t) > \frac{1}{(1 - \eta_t)} \gamma_t \rho^m \langle \nabla f(Z_t), D_t \rangle.$$

According to the mean-theorem, there is a $\theta_t \in (0, 1)$ such that

$$s\rho^m \langle \nabla f(Z_t + \theta_t s\rho^m D_t), D_t \rangle > \frac{1}{(1 - \eta_t)} \gamma_t \rho^m \langle \nabla f(Z_t), D_t \rangle,$$

that is,

$$\langle \nabla f(Z_t + \theta_t \rho^m D_t) - \nabla f(Z_t), D_t \rangle > (\frac{\gamma_t}{s(1 - \eta_t)} - 1) \langle \nabla f(Z_t), D_t \rangle.$$

When $m \to \infty$, we find that

$$(\frac{\gamma_t}{s(1 - \eta_t)} - 1) \langle \nabla f(Z_t), D_t \rangle \leq 0.$$

Since $0 < \frac{\gamma_t}{1 - \eta_t} < 1 < s$, $\langle \nabla f(Z_t), D_t \rangle \geq 0$ is correct. This is not consistent with the fact that $\langle \nabla f(Z_t), D_t \rangle \leq 0$. Therefore, step 5 of Algorithm 1 is well-defined. □

3. Convergence Analysis

In this part, we prove the global convergence of NMPBB. To establish the global convergence of NMPBB, we firstly present the following result.

Lemma 7. *Suppose that Algorithm 1 generates a step size λ_t, if the stationary point of (3) is not W_{t+1}, so there is a constant $\tilde{\lambda}$ that will cause $\lambda_t \geq \tilde{\lambda}$.*

Proof. For the resulting step size λ_t, if λ_t does not satisfy (26), namely,

$$\begin{aligned} f(Z_t + s\lambda_t D_t) &> S_t + \frac{1}{1-\eta_t}\gamma_t\lambda_t[\langle \nabla f(Z_t), D_t\rangle + \frac{\mu}{\alpha_t}\|D_t\|^2] \\ &\geq S_t + \frac{1}{1-\eta_t}\gamma_t\lambda_t\langle \nabla f(Z_t), D_t\rangle \\ &\geq f(Z_t) + \frac{1}{1-\eta_t}\gamma_t\lambda_t\langle \nabla f(Z_t), D_t\rangle \end{aligned}$$

where Lemmas 5 and 6 lead to the final inequality. Thus,

$$f(Z_t + s\lambda_t D_t) - f(Z_t) \geq \frac{1}{1-\eta_t}\gamma_t\lambda_t\langle \nabla f(Z_t), D_t\rangle. \tag{42}$$

By the mean-value theorem, we can find an $\theta \in (0,1)$ that makes

$$\begin{aligned} f(Z_t + s\lambda_t D_t) - f(Z_t) &= s\lambda_t\langle \nabla f(Z_t + \theta s\lambda_t D_t), D_t\rangle \\ &= s\lambda_t\langle \nabla f(Z_t), D_t\rangle + s\lambda_t\langle \nabla f(Z_t + \theta_t s\lambda_t D_t) \\ &\quad - \nabla f(Z_t), D_t\rangle \\ &\leq s\lambda_t\langle \nabla f(Z_t), D_t\rangle + s^2 L_W \lambda_t^2\|D_t\|^2, \end{aligned} \tag{43}$$

where $L_W > 0$ is the Lipschitz constant of $\nabla f(W_t)$.

Substitute the last inequality we obtained from (43) into (42) to find

$$\lambda_t \geq \frac{s(1-\eta_t) - \gamma_t}{L_W s^2 \alpha_{max}(1-\eta_t)}. \tag{44}$$

From $\eta_{t-1} \in [\eta_{min}, \eta_{max}]$ and $\gamma_t \in [\gamma_{min}, \gamma_{max}(1-\eta_{max})]$, we have

$$\lambda_t \geq \frac{s(1-\eta_{max}) - \gamma_{max}}{L_W s^2 \alpha_{max}(1-\eta_{min})} := \tilde{\lambda}. \tag{45}$$

□

Lemma 8. *Assume that Algorithm 1 generates the sequence $\{W_t\}$, for the given level set $\mathcal{L}(W_0)$, if it is considered bounded, so there is*

(i)

$$\lim_{t\to\infty} S_t = \lim_{t\to\infty} f(W_t). \tag{46}$$

(ii) *there is a positive constant δ makes*

$$S_t - f(W_{t+1}) \geq \delta\|D_{t+1}\|^2. \tag{47}$$

Proof. (i) By the definition of S_{t+1}, for $t \geq 1$ we have

$$S_{t+1} - S_t = (1-\eta_t)(f(W_{t+1}) - S_t).$$

Since $\eta_{max} \in [0,1]$, and $\eta_t \in [\eta_{min}, \eta_{max}]$ for all t,

$$1 - \eta_{min} \geq 1 - \eta_t \geq 1 - \eta_{max} > 0.$$

According to Theorem 1, as $t \to \infty$,

$$\lim_{t\to\infty} \frac{1}{1-\eta_{max}}(S_{t+1} - S_t) = \lim_{t\to\infty} \frac{1}{1-\eta_{min}}(S_{t+1} - S_t) = 0. \tag{48}$$

which implies that

$$\lim_{t\to\infty}(f(W_{t+1}) - S_t) = 0. \tag{49}$$

(ii) From (11) and Lemma 2 (i), we have

$$\begin{aligned} S_t - f(W_{t+1}) &\geq -\frac{1}{1-\eta_t}\gamma_t\lambda_t[\langle\nabla f(Z_t), D_t\rangle + \frac{\mu}{\alpha_t}\|D_t\|^2] \\ &\geq \frac{\gamma_{min}}{1-\eta_{min}}\frac{\lambda_t}{\alpha_t}(1-\mu)\|D_t\|^2 \\ &\geq \frac{\gamma_{min}\tilde{\lambda}(1-\mu)}{(1-\eta_{min})\alpha_{max}}\|D_t\|^2 \\ &= \delta\|D_t\|^2, \end{aligned} \tag{50}$$

where $\delta = \frac{\gamma_{min}\tilde{\lambda}(1-\mu)}{(1-\eta_{min})\alpha_{max}}$. □

The global convergence of Algorithm 1 is proved by the theorem shown below.

Theorem 3. *Suppose that Algorithm 1 generates sequences $\{Z_t\}$ and $\{W_t\}$, so we obtain*

$$\lim_{t\to\infty}\|D_t\| = 0. \tag{51}$$

Proof. According to Lemma 8 (ii), we have

$$S_t - f(W_{t+1}) \geq \delta\|D_t\|^2 \geq 0 \ \forall t \in \mathbf{N}.$$

Based on Lemma 8 (i), as $t \to \infty$, we can obtain

$$\lim_{t\to\infty}\|D_t\| = 0.$$

□

According to Theorem 3, Lemma 3, and (25), we will exhibit the main convergence results we find as follows.

Theorem 4. *For a given level set $\mathcal{L}(W_0)$, assume that it is bounded, hence Algorithm 1 computes the generated sequence $\{W_t\}$, and any accumulation point obtained is a stationary point of (3).*

4. Numerical Experiments

In the following content, by using synthetic datasets and real-world datasets (ORL image database and Yale image database (Both ORL and Yale image datasets in MATLAB format are available at http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html (accessed on 26 December 2023))), we exhibit the main numerical experiments to compare the performance of NMPBB with that of the other five efficient methods including the NeNMF [29], the projected BB method (APBB2 [17]) (The code is available at http://homepages.umflint.edu/$\sim$lxhan/software.html (accessed on 26 December 2023)), QRPBB [18], hierarchical alternating least squares (HALS) [38], and block coordinate descent (BCD) method [39]. All of the reported numerical results are performed using MATLAB v8.1 (R2013a) on a Lenovo laptop.

4.1. Stopping Criterion

According to the Karush-Kuhn-Tucker (KKT) conditions optimized by existing constraints, we know that (W^k, H^k) is a stationary point of NMF (2) if and only if $\nabla_W^P f(W,H) = 0$ and $\nabla_H^P f(W,H) = 0$ are simultaneously satisfied, here

$$[\nabla_W^P f(W,H)]_{ij} = \begin{cases} [\nabla_W f(W,H)]_{ij}, & \text{if } W_{ij} > 0, \\ \min\{0, [\nabla_W f(W,H)]_{ij}\}, & \text{if } W_{ij} = 0, \end{cases}$$

and $\nabla_H^P f(W^{(k)}, H^{(k)})$ is also written as shown above. Hence, we employ the stopping criteria shown below, which is also used in [40] in numerical experiments:

$$\|[\nabla_W^P f(W^{(k)}, H^{(k)}), \nabla_H^P f(W^{(k)}, H^{(k)})^T]\| \tag{52}$$
$$\leq \epsilon \|[\nabla_W^P f(W^{(1)}, H^{(1)}), \nabla_H^P f(W^{(1)}, H^{(1)})^T]\|, \tag{53}$$

here $\epsilon > 0$ is a tolerance. When employing the stop criterion (52), we need to pay attention to the scale degrees of freedom of the NMF solution, as discussed in [41].

4.2. Synthetic Data

In this section, first the NMPBB method and the other three ANLS-based methods are tested on synthetic datasets. Since the matrix V in this test happens to be a low-rank matrix, it will be rewritten as $V = LR$, and here we generate the L and R by using the MATLAB commands $max(0, randn(m,r))$ and $max(0, randn(r,n))$, respectively.

For NMPBB, in a later experiment we adopt the parameters shown below:

$$\alpha_{max} = 10^{20}, \alpha_{min} = 10^{-20}, \rho = 0.25, \gamma = 10^{-3}.$$

The settings are identical with those of APBB2 and QRPBB. Take $s = 1.7$ for NMPBB, the reason of selecting relaxation factor $s = 1.7$ is given in Section 4.4, and take $tol = 10^{-8}$ for all comparison algorithms. In addition, for NMPBB we choose $\eta_0 = 0.15$ and the update η_t by the following recursive formula

$$\eta_t = \begin{cases} \frac{\eta_0}{2}, & \text{if } t = 1, \\ \frac{\eta_{t-1} + \eta_{t-2}}{2}, & \text{if } t \geq 2. \end{cases}$$

We unify the maximum number of iterations of all algorithms to 50,000. All other parameters of APBB2, NeNMF, and QRPBB are unified as default values.

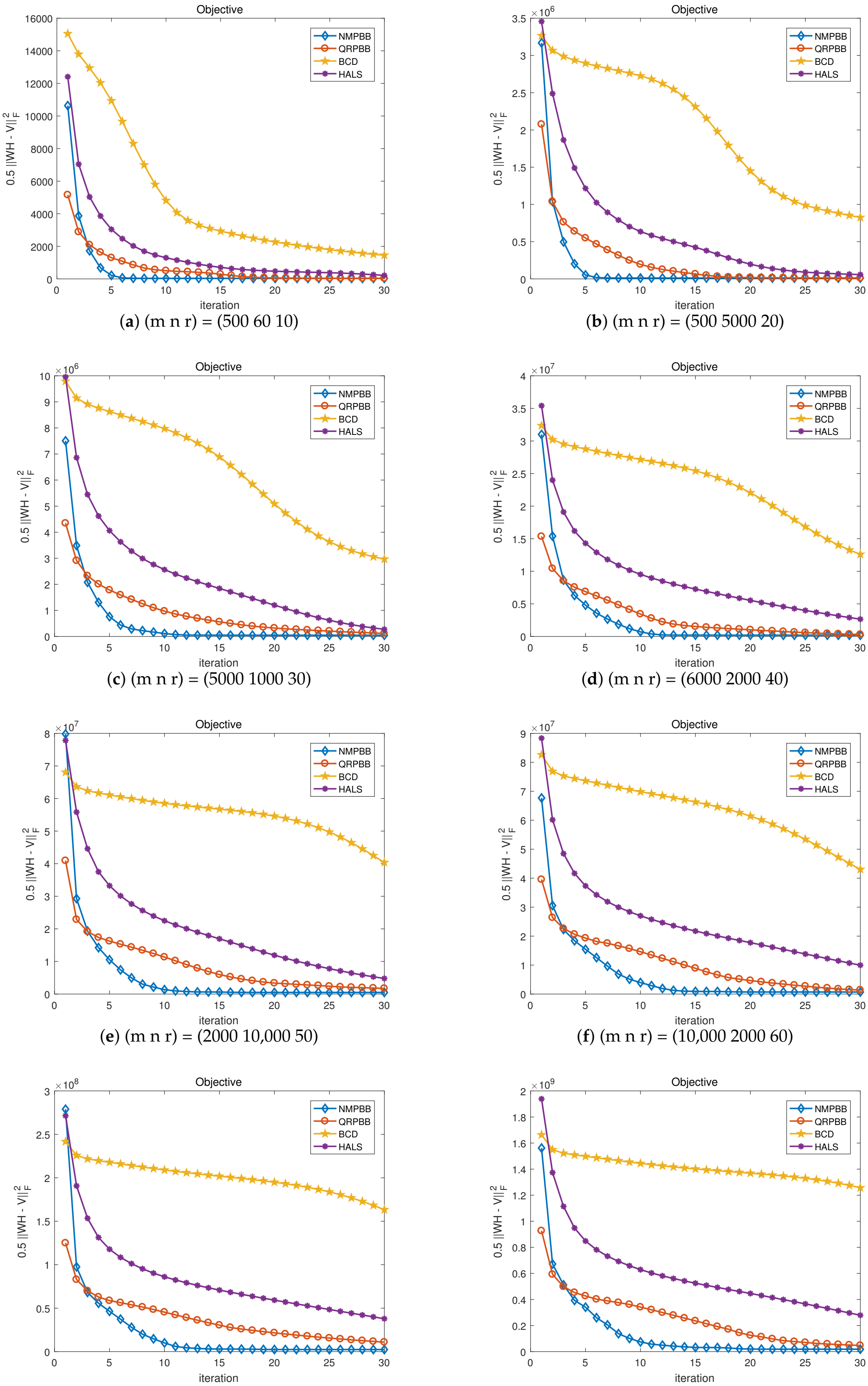
For all the problems we are considering, casually generated 10 diverse starting values, and the average outcomes obtained from using these starting points are presented in Table 1. The item iter represents that the number of iterations required to satisfy the termination condition (52) is met. The item niter represents the total number of sub-iterations for solving W and H. $\|V - W^k H^k\|_F / \|V\|_F$ is relative error, $\|[\nabla_H^P f(W^k, H^k), \nabla_W^P f(W^k, H^k)]\|_F$ is the final value of the projected gradient norm, and CPU time (in seconds) separately measures performance.

Table 1 clearly indicates that all methods met the condition of convergence within a reasonable number of iterations. Table 1 also clearly indicates that our ANMPBB needs the least execution time and the least number of sub-iterations among all methods, particularly in the case of large-scale problems.

Since the NMPBB method is closely related to the QRPBB method, as we all know that the hierarchical ALS (HALS) algorithm for NMF is the most effective upon most occasions, we use the coordinate descent method to solve subproblems in NMF. We further examine algorithms of NMPBB, QRPBB, HALS, and BCD. We show that these four methods compare on eight randomly generated independent Gaussian noise measures when the signal-to-noise ratio which is 30 dB in Figures 2–4 is terminated when the stopping criterion said by the inequality in (52) satisfies $\epsilon = 10^{-8}$ or the maximum number of iterations is more than 30. Figure 2 shows the value of the objective function compared to the number of iterations. From Figure 2, for most of the test problems, we will draw a conclusion that NMPBB decreases the objective function much quicker than the other three methods in 30 iterations. This may be because our NMPBB exploits an efficient modified nonmonotone line search and adds a relaxing factor s to the update rules of W_{t+1} and H_{t+1}. Hence our NMPBB significantly outperforms the other three methods. Figure 3 shows the relationship between the relative residual errors and the number of iterations. Figure 4 exhibits the relative residual errors versus CPU time. The results shown in Figures 3 and 4 are consistent with those shown in Figure 2.

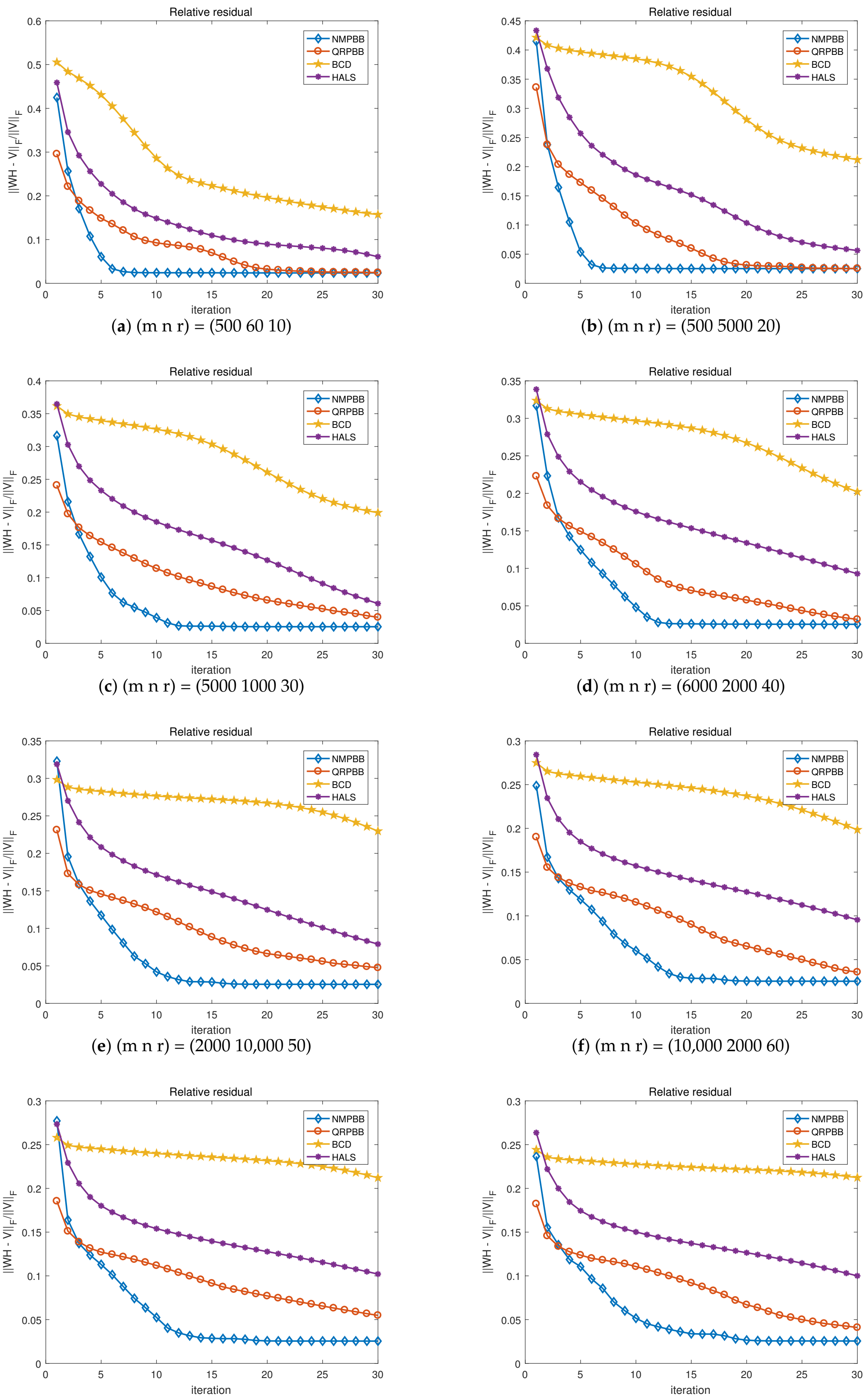
Table 1. Experimental results on synthetic datasets.

(m n r)	Alg	Iter	Niter	Pgn	Time	Residual
(200 100 10)	NeNMF	153.3	6073.7	3.44×10^{-5}	0.25	0.4596
	APBB2	171.9	2442.8	2.76×10^{-5}	0.26	0.4596
	QRPBB	158.0	1476.4	2.66×10^{-5}	0.19	0.4596
	NMPBB	50.3	496.4	2.50×10^{-5}	0.09	0.4596
(100 500 20)	NeNMF	1946.7	83,561.7	1.62×10^{-4}	14.46	0.4257
	APBB2	2798.7	48,444.2	1.31×10^{-4}	15.77	0.4257
	QRPBB	2365.7	26,052.7	1.32×10^{-4}	8.49	0.4258
	NMPBB	625.4	7400.4	1.31×10^{-4}	2.67	0.4257
(500 300 25)	NeNMF	687.3	28,304.9	3.73×10^{-4}	7.30	0.4496
	APBB2	456.5	8077.3	3.20×10^{-4}	5.00	0.4496
	QRPBB	436.6	5452.2	3.26×10^{-4}	3.31	0.4496
	NMPBB	135.1	1958.1	2.77×10^{-4}	1.46	0.4496
(700 700 30)	NeNMF	183.4	6638.0	1.04×10^{-3}	3.45	0.4588
	APBB2	161.5	3438.7	8.83×10^{-4}	4.56	0.4588
	QRPBB	153.0	2191.9	9.11×10^{-4}	2.78	0.4588
	NMPBB	60.7	936.4	8.41×10^{-4}	1.05	0.4588
(1000 500 30)	NeNMF	221.0	7685.5	1.05×10^{-3}	4.22	0.4578
	APBB2	180.4	3513.8	8.62×10^{-4}	4.52	0.4578
	QRPBB	162.8	2195.5	9.41×10^{-4}	2.63	0.4578
	NMPBB	60.5	937.4	9.17×10^{-4}	1.50	0.4578
(600 1000 40)	NeNMF	1139.0	43,519.6	1.69×10^{-3}	33.86	0.4515
	APBB2	554.4	9117.8	1.40×10^{-3}	18.90	0.4515
	QRPBB	434.2	5963.0	1.52×10^{-3}	9.69	0.4515
	NMPBB	143.4	2489.9	1.22×10^{-3}	3.77	0.4515
(1000 600 40)	NeNMF	644.5	25,379.5	1.68×10^{-3}	20.00	0.4518
	APBB2	723.3	12,948.1	1.41×10^{-3}	26.16	0.4518
	QRPBB	536.5	7686.2	1.31×10^{-3}	12.55	0.4518
	NMPBB	137.8	2262.7	1.18×10^{-3}	3.53	0.4518
(1000 2000 50)	NeNMF	330.8	12,081.3	4.98×10^{-3}	25.35	0.4574
	APBB2	240.3	4783.6	4.29×10^{-3}	23.41	0.4574
	QRPBB	252.8	4264.2	3.84×10^{-3}	18.29	0.4574
	NMPBB	79.1	1558.7	4.10×10^{-3}	6.12	0.4574
(2000 2000 50)	NeNMF	172.3	6796.9	8.25×10^{-3}	18.96	0.4629
	APBB2	147.6	3734.1	7.30×10^{-3}	24.92	0.4629
	QRPBB	149.0	2524.7	5.83×10^{-3}	16.43	0.4629
	NMPBB	57.1	1089.3	5.75×10^{-3}	6.81	0.4629
(3000 1000 60)	NeNMF	485.7	17,642.4	8.79×10^{-3}	63.10	0.4555
	APBB2	396.3	7386.3	7.29×10^{-3}	64.50	0.4555
	QRPBB	380.3	6049.4	6.77×10^{-3}	48.12	0.4555
	NMPBB	116.2	2141.4	5.81×10^{-3}	16.55	0.4555
(5000 1000 70)	NeNMF	1036.9	50,207.5	1.65×10^{-2}	344.92	0.4540
	APBB2	1397.7	23,570.8	1.47×10^{-2}	433.55	0.4540
	QRPBB	1307.3	20,456.8	1.36×10^{-2}	304.93	0.4540
	NMPBB	281.7	5639.0	1.22×10^{-2}	76.28	0.4540



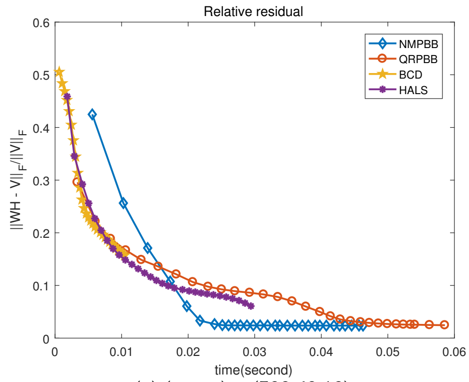
(**a**) (m n r) = (500 60 10)
(**b**) (m n r) = (500 5000 20)
(**c**) (m n r) = (5000 1000 30)
(**d**) (m n r) = (6000 2000 40)
(**e**) (m n r) = (2000 10,000 50)
(**f**) (m n r) = (10,000 2000 60)
(**g**) (m n r) = (10,000 5000 70)
(**h**) (m n r) = (30,000 10,000 80)

Figure 2. Objective value versus iteration on random problem $\min_{W,H\geq 0} \frac{1}{2}\|V - WH\|_F^2$.

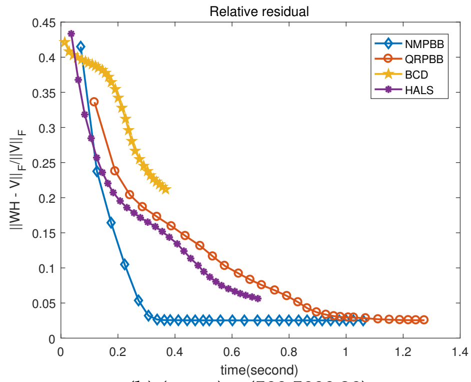


(**a**) (m n r) = (500 60 10) (**b**) (m n r) = (500 5000 20)
(**c**) (m n r) = (5000 1000 30) (**d**) (m n r) = (6000 2000 40)
(**e**) (m n r) = (2000 10,000 50) (**f**) (m n r) = (10,000 2000 60)
(**g**) (m n r) = (10,000 5000 70) (**h**) (m n r) = (30,000 10,000 80)

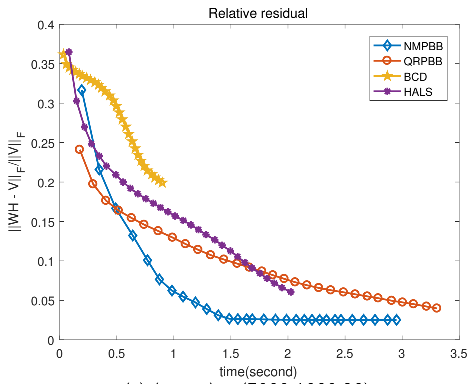
Figure 3. Residual value versus iteration on random problem $\min_{W,H\geq 0} \frac{1}{2}\|V - WH\|_F^2$.



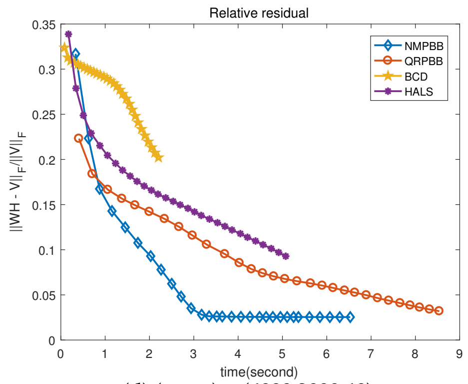
(**a**) (m n r) = (500 60 10)



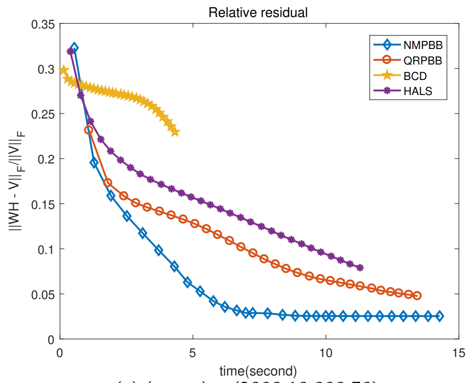
(**b**) (m n r) = (500 5000 20)



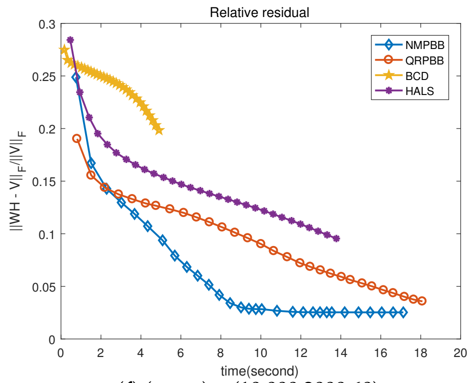
(**c**) (m n r) = (5000 1000 30)



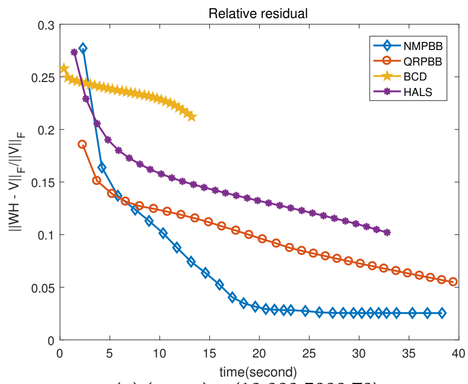
(**d**) (m n r) = (6000 2000 40)



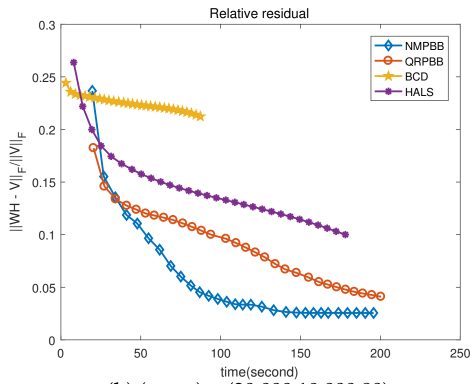
(**e**) (m n r) = (2000 10,000 50)



(**f**) (m n r) = (10,000 2000 60)



(**g**) (m n r) = (10,000 5000 70)



(**h**) (m n r) = (30,000 10,000 80)

Figure 4. Residual value versus CPU time on random problem $\min_{W,H\geq 0} \frac{1}{2}\|V - WH\|_F^2$.

4.3. Image Data

The ORL image database is a collection of 400 images of people's faces belonging to 40 individuals representing 10 each. The dataset includes variations in lighting conditions, facial expressions (including whether they open their eyes, whether they smile), and facial details including whether they wear glasses. Some subjects have multiple photos taken at different times. The images were captured with the subject positioned upright and facing forward (allowing for slight movement to the sides). The background used was uniformly dark and even. All the images were taken against a dark homogeneous background with the subjects in an upright frontal position (with tolerance for some side movement). The pictures used are represented by the columns of the matrix V, and V has 400 rows and 1024 columns.

The Yale face database was created at the Yale Center for Computational Vision and Control. It consists of 165 gray-scale images, with each person in the database having 11 images associated with them. In total, there are 15 people. The facial images in question were captured under different lighting conditions (left-light, center-light, right-light), with various facial expressions (calm, cheerful, sorrowful, amazed, and blinking), and with or without glasses. The pictures used are represented by the rows of the matrix V, and V has 165 rows and 1024 columns.

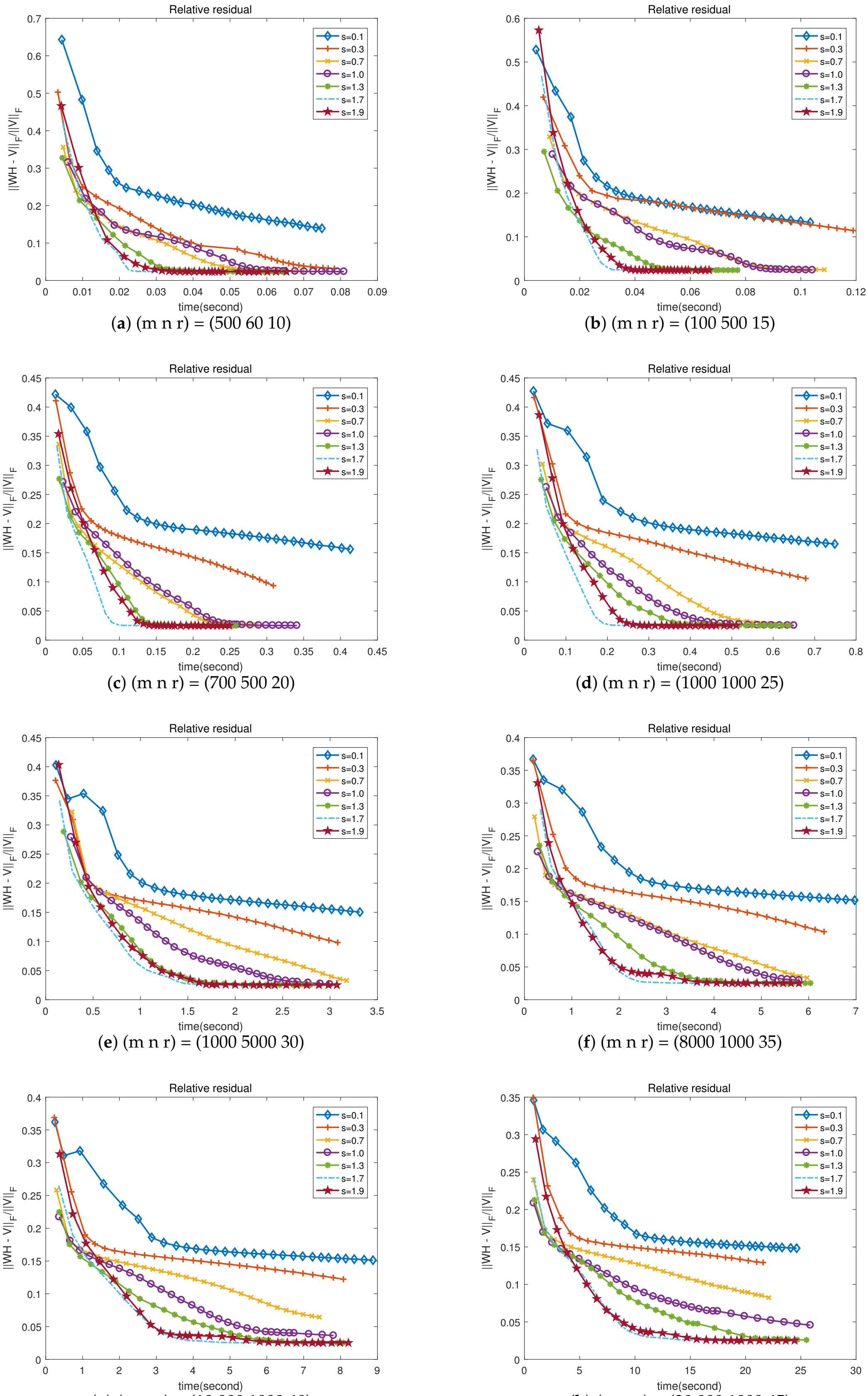
For all the databases we used in (52), we performed a diverse casually generated starting iteration with $\epsilon = 10^{-8}$, the maximum number of iterations (maxit) for all algorithms is set to 50,000, and the average results are presented in Table 2. From Table 2, we conclude that the QRPBB method converges in fewer iterations and CPU times than APBB2 and NeNMF, and in contrast to QRPBB, our NMPBB method requires $1/4$ CPU time to satisfy the set tolerance. Although the residuals by NMPBB are not the smallest among all algorithms appearing for all the databases we use, the results of pgn mean that solutions by NMPBB are nearer to the point of stationary.

Table 2. Experimental results on Yale and ORL datasets.

(m n r)	Alg	Iter	Niter	Pgn	Time	Residual
(165 1024 25)	NeNMF	3735.1	178,254.1	4.41×10^{-1}	65.78	0.1930
	APBB2	3079.6	97,375.7	6.42×10^{-2}	78.75	0.1930
	QRPBB	2711.1	54,215.7	6.16×10^{-2}	42.25	0.1931
	NMPBB	1019.2	24,063.1	2.60×10^{-2}	16.57	0.1930
(400 1024 25)	NeNMF	13,613.4	836,034.3	7.71×10^{-2}	349.62	0.1117
	APBB2	9430.6	446,361.6	6.88×10^{-2}	474.26	0.1117
	QRPBB	7593.5	213,178.5	7.05×10^{-2}	205.26	0.1117
	NMPBB	1982.7	41,597.0	6.25×10^{-2}	34.22	0.1117

4.4. The Importance of Relaxation Factor s

In the following content, the clear experimental results indicate that relaxation factor s is used for updating rules of W_{t+1} and H_{t+1}. We implement NMPBB using diverse s given $s = 0.1, 0.3, 0.7, 1.0, 1.3, 1.7, 1.9$ on synthetic datasets which are the same as those in Section 4.2. We set the required maximum number of iterations to 30, and the other parameters required in the experiment will have the same values as those in Section 4.2. Figure 5 shows the relationship between the relative residuals error and the run-time results. In Figure 5, we can see that the relaxation factor s fails to accelerate the convergence when $s < 1$ and increasing constant s significantly accelerates the convergence when $1 < s < 2$. As for NMPBB, it seems that s = 1.7 is the best compared with other experimental values in terms of speed of convergence, and hence s = 1.7 was used as our NMPBB in all experiments.



(**a**) (m n r) = (500 60 10)
(**b**) (m n r) = (100 500 15)
(**c**) (m n r) = (700 500 20)
(**d**) (m n r) = (1000 1000 25)
(**e**) (m n r) = (1000 5000 30)
(**f**) (m n r) = (8000 1000 35)
(**g**) (m n r) = (10,000 1000 40)
(**h**) (m n r) = (30,000 1000 45)

Figure 5. Residual value versus CPU time on random problem $\min_{W,H\geq 0} \frac{1}{2}\|V - WH\|_F^2$.

5. Conclusions

In this paper, a prox-linear quadratic regularization objective function is presented, and the prox-linear term leads to strongly convex quadratic subproblems. Then, we propose a new line search technique based on the idea of [33]. According to the new line search, we put forward a global convergent method with larger step size to solve the subproblems. Finally, a series of numerical results are given to show that the method is a promising tool for NMF.

Symmetric nonnegative matrix factorization is a special but important class of NMF which has found numerous applications in data analysis such as various clustering tasks. Therefore, a direction for future research would be to extend the proposed algorithm to solve symmetric nonnegative matrix factorization problems.

Author Contributions: W.L.: supervision, methodology, formal analysis, writing—original draft, writing—review and editing. X.S.: software, data curation, conceptualization, visualization, formal analysis, writing—original draft. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China under grant No. 12201492.

Data Availability Statement: The datasets generated or analyzed during this study are available in the face databases in matlab format at http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html (accessed on 26 December 2023).

Conflicts of Interest: The authors declare that they have no conflicts of interest.

References

1. Gong, P.H.; Zhang, C.S. Efficient nonnegative matrix factorization via projected Newton method. *Pattern Recognit.* **2012**, *45*, 3557–3565. [CrossRef]
2. Lee, D.D.; Seung, H.S. Learning the parts of objects by non-negative matrix factorization. *Nature* **1999**, *401*, 788–791. [CrossRef] [PubMed]
3. Lee, D.D.; Seung, H.S. Algorithms for non-negative matrix factorization. *Adv. Neural Process. Inf. Syst.* **2001**, *13*, 556–562.
4. Kim, D.; Sra, S.; Dhillon, I.S. Fast Newton-type methods for the least squares nonnegative matrix approximation problem. *SIAM Int. Conf. Data Min.* **2007**, *1*, 38–51.
5. Paatero, P.; Tapper, U. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics* **1994**, *5*, 111–126. [CrossRef]
6. Ding, C.; Li, T.; Peng, W. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Comput. Stat. Data Anal.* **2008**, *52*, 3913–3927. [CrossRef]
7. Chan, T.H.; Ma, W.K.; Chi, C.Y.; Wang, Y. A convex analysis framework for blind separation of nonnegative sources. *IEEE Trans. Signal Process.* **2008**, *56*, 5120–5134. [CrossRef]
8. Ding, C.; He, X.; Simon, H. On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering. *SIAM Int. Conf. Data Min. (SDM'05)* **2005**, 606–610.
9. Févotte, C.; Bertin, N.; Durrieu, J.L. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. *Neural Comput.* **2009**, *21*, 793–830.
10. Ma, W.K.; Bioucas-Dias, J.; Chan, T.H.; Gillis, N.; Gader, P.; Plaza, A.; Ambikapathi, A.; Chi, C.Y. A Signal Processing Perspective on Hyperspectral Unmixing. *IEEE Signal Process. Mag.* **2014**, *31*, 67–81. [CrossRef]
11. Barzilai, J.; Borwein, J.M. Two-point step size gradient methods. *IMA J. Numer. Anal.* **1988**, *8*, 141–148. [CrossRef]
12. Dai, Y.H.; Liao, L.Z. R-Linear convergence of the Barzilai-Borwein gradient method. *IMA J. Numer. Anal.* **2002**, *22*, 1–10. [CrossRef]
13. Raydan, M. On the Barzilai-Borwein choice of steplength for the gradient method. *IMA J. Numer. Anal.* **1993**, *13*, 321–326. [CrossRef]
14. Raydan, M. The Barzilai and Borwein gradient method for the large-scale unconstrained minimization problem. *SIAM J. Optim.* **1997**, *7*, 26–33. [CrossRef]
15. Xiao, Y.H.; Hu, Q.J. Subspace Barzilai-Borwein gradient method for large-scale bound constrained optimization. *Appl. Math. Optim.* **2008**, *58*, 275–290. [CrossRef]
16. Xiao, Y.H.; Hu, Q.J.; Wei, Z.X. Modified active set projected spectral gradient method for bound constrained optimization. *Appl. Math. Model.* **2011**, *35*, 3117–3127. [CrossRef]
17. Han, L.X.; Neumann, M.; Prasad, U. Alternating projected Barzilai-Borwein methods for nonnegative matrix factorization. *Electron. Trans. Numer. Anal.* **2009**, *36*, 54–82.

18. Huang, Y.K.; Liu, H.W.; Zhou, S.S. Quadratic regularization projected alternating Barzilai-Borwein method for nonnegative matrix factorization. *Data Min. Knowl. Discov.* **2015**, *29*, 1665–1684. [CrossRef]
19. Huang, Y.K.; Liu, H.W.; Zhou, S.S. An efficint monotone projected Barzilai-Borwein method for nonnegative matrix factorization. *Appl. Math. Lett.* **2015**, *45*, 12–17. [CrossRef]
20. Li, X.L.; Liu, H.W.; Zheng, X.Y. Non-monotone projection gradient method for non-negative matrix factorization. *Comput. Optim. Appl.* **2012**, *51*, 1163–1171. [CrossRef]
21. Liu, H.W.; Li, X. Modified subspace Barzilai-Borwein gradient method for non-negative matrix factorization. *Comput. Optim. Appl.* **2013**, *55*, 173–196. [CrossRef]
22. Bonettini, S. Inexact block coordinate descent methods with application to non-negative matrix factorization. *IMA J. Numer. Anal.* **2011**, *31*, 1431–1452. [CrossRef]
23. Zdunek, R.; Cichocki, A. Fast nonnegative matrix factorization algorithms using projected gradient approaches for large-scale problems. *Comput. Intell. Neurosci.* **2008**, *2008*, 939567. [CrossRef]
24. Bai, J.C.; Bian, F.M.; Chang, X.K.; Du, L. Accelerated stochastic Peaceman-Rachford method for empirical risk minimization. *J. Oper. Res. Soc. China* **2023**, *11*, 783–807. [CrossRef]
25. Bai, J.C.; Han, D.R.; Sun, H.; Zhang, H.C. Convergence on a symmetric accelerated stochastic ADMM with larger stepsizes. *CSIAM Trans. Appl. Math.* **2022**, *3*, 448–479.
26. Bai, J.C.; Hager, W.W.; Zhang, H.C. An inexact accelerated stochastic ADMM for separable convex optimization. *Comput. Optim. Appl.* **2022**, *81*, 479–518. [CrossRef]
27. Bai, J.C.; Li, J.C.; Xu, F.M.; Zhang, H.C. Generalized symmetric ADMM for separable convex optimization. *Comput. Optim. Appl.* **2018**, *70*, 129–170. [CrossRef]
28. Bai, J.C.; Zhang, H.C.; Li, J.C. A parameterized proximal point algorithm for separable convex optimization. *Optim. Lett.* **2018**, *12*, 1589–1608. [CrossRef]
29. Guan, N.Y.; Tao, D.C.; Luo, Z.G.; Yuan, B. NeNMF: An optimal gradient method for nonnegative matrix factorization. *IEEE Trans. Signal Process.* **2012**, *60*, 2882–2898. [CrossRef]
30. Xu, Y.Y.; Yin, W.T. A globally convergent algorithm for nonconvex optimization based on block coordinate update. *J. Sci. Comput.* **2017**, *72*, 700–734. [CrossRef]
31. Zhang, H.C.; Hager, W.W. A nonmonotone line search technique and its application to unconstrained optimization. *SIAM J. Optim.* **2004**, *14*, 1043–1056. [CrossRef]
32. Dai, Y.H. On the nonmonotone line search. *J. Optim. Theory Appl.* **2002**, *112*, 315–330. [CrossRef]
33. Gu, N.Z.; Mo, J.T. Incorporating nonmonotone strategies into the trust region method for unconstrained optimization. *Comput. Math. Appl.* **2008**, *55*, 2158–2172. [CrossRef]
34. Ahookhosh, M.; Amini, K.; Bahrami, S. A class of nonmonotone Armijo-type line search method for unconstrained optimization. *Optimization* **2012**, *61*, 387–404. [CrossRef]
35. Nosratipour, H.; Borzabadi, A.H.; Fard, O.S. On the nonmonotonicity degree of nonmonotone line searches. *Calcolo* **2017**, *54*, 1217–1242. [CrossRef]
36. Glowinski, R. *Numerical Methods for Nonlinear Variational Problems*; Springer: New York, NY, USA, 1984.
37. Birgin, E.G.; Martinez, J.M.; Raydan, M. Nonmonotone spectral projected gradient methods on convex sets. *SIAM J. Optim.* **2000**, *10*, 1196–1211. [CrossRef]
38. Cichocki, A.; Zdunek, R.; Amari, S.I. Hierarchical ALS Algorithms for Nonnegative Matrix and 3D Tensor Factorization. *Lect. Notes Comput. Sci. Springer* **2007**, *4666*, 169–176.
39. Xu, Y.Y.; Yin, W.T. A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion. *SIAM J. Imaging Sci.* **2015**, *6*, 1758–1789. [CrossRef]
40. Lin, C.J. Projected Gradient Methods for non-negative matrix factorization. *Neural Comput.* **2007**, *19*, 2756–2779. [CrossRef] [PubMed]
41. Gillis, N. The why and how of nonnegative matrix factorization. *arXiv* **2015**, arXiv:1401.5226v2.