



Research article

Adaptive moments method for linear hyperspectral unmixing

Zhewei Zhang and Fangfang Xu*

College of Mathematics and Systems Science, Shandong University of Science and Technology,
Qingdao, Shandong 266590, China

* **Correspondence:** Email: xuff@sdust.edu.cn.

Abstract: Linear hyperspectral unmixing (LHU) aims to decompose mixed pixels into endmembers and abundance vectors, and it has important applications in terrain classification, mineral recognition and quantification, agricultural monitoring, and military surveillance. Because the data-fidelity term couples the endmember and abundance variables, the original formulation is nonconvex. After a principal components analysis (PCA)-based dimensionality reduction and the inverse transformation $Q = M^{-1}$, we obtain a smooth reduced model on the nonsingular domain. We first apply the adaptive moments method (ADAM) to this transformed LHU model and then develop a stochastic adaptive moments method (SADAM) based on mini-batch gradients to reduce the computational cost for large data sets. For SADAM, we provide a conditional convergence analysis to stationary points under smoothness, boundedness, and moment-tracking assumptions. Numerical experiments on four real hyperspectral data sets show that SADAM can reduce computational time while maintaining competitive unmixing accuracy.

Keywords: linear hyperspectral unmixing; adaptive moments method; stochastic adaptive moments method; coupled objective function; convergence rate

Mathematics Subject Classification: 65K10, 90C30, 68U10

1. Introduction

Hyperspectral images have many advantages. However, the existence of mixed pixels seriously affects the accurate interpretation of hyperspectral images. Linear hyperspectral unmixing (LHU) is an effective technique for extracting endmembers from mixed pixels. Therefore, LHU is widely used in terrain classification, mineral recognition and quantification, agricultural monitoring, and military surveillance [1].

LHU assumes that the spectral vectors are in the convex body of endmembers. Let $Y = [y_1, \dots, y_n] \in \mathbb{R}^{B \times n}$ be the matrix composed of mixing spectral vectors y_i . $M = [m_1, \dots, m_p] \in \mathbb{R}^{B \times p}$ is the mixing matrix containing the endmembers. S is the abundance matrix, whose column vectors $s_i (i = 1, \dots, n)$ belong to the standard p -simplex set $\mathcal{S}_p = \{s \in \mathbb{R}^p : s \geq \mathbf{0}, \mathbf{1}_p^\top s = 1\}$. Based on the above assumption, the relationship between Y and M is

$$Y = MS. \quad (1.1)$$

Because noise is unavoidable in practice, the equality constraint is relaxed and penalized in the objective function. Because M and S are both unknown, the penalty term $\|Y - MS\|_F^2$ is a nonconvex function. Moreover, the number of rows of Y and M is equal to the number of spectral bands B , which is much larger than the number of endmembers p . The unknown variable M is an ill-conditioned matrix, which is difficult to estimate.

Based on the above two difficulties, a preprocessing step of Y is usually conducted by principal components analysis (PCA). PCA can preserve the most important features of data and remove unimportant features [2].

PCA can be conducted by two computational steps. First of all, PCA constitutes an approximate subspace for the row space of Y . Projecting Y into a subspace of a set of bases reduces the row space of Y from B dimension to p dimension. Secondly, Y and M are projected to the subspace. After preprocessing, $Y \in \mathbb{R}^{p \times n}$ has p rows, and $M \in \mathbb{R}^{p \times p}$ becomes a square matrix in the reduced space. Under the standard assumption that the projected endmembers are linearly independent, M is nonsingular. Accordingly, we work on the domain

$$\Omega = \{Q \in \mathbb{R}^{p \times p} : \det(Q) \neq 0\},$$

on which Q^{-T} and $\log |\det(Q)|$ are well defined. Let $Q = M^{-1}$; then the relationship between mixed pixels and endmembers can also be expressed as

$$QY = S. \quad (1.2)$$

The aim of transforming Eq (1.1) into (1.2) is to decouple the variables M and S . In addition, the volume of the simplex formed by the endmembers is proportional to $|\det(M)|$, and $\log |\det(M)| = -\log |\det(Q)|$ holds because $Q = M^{-1}$. Moreover, in view of the presence of noise, the equality constraint is relaxed and penalized in the objective function. The model of LHU is considered,

$$\begin{aligned} \min_{Q,S} \quad & \frac{1}{2} \|QY - S\|_F^2 - \lambda \log |\det(Q)|, \\ \text{s.t.} \quad & S \geq \mathbf{0}_{p \times n}, \quad \mathbf{1}_p^\top S = \mathbf{1}_n^\top, \end{aligned} \quad (1.3)$$

where $\lambda > 0$ weighs the simplex-volume penalty against the reconstruction error, and $\mathbf{1}_n$ denotes the n -dimensional vector of all ones. The objective function of model (1.3) combines the advantages of geometry-based and statistics-based approaches, but it is not jointly convex in (Q, S) .

To make the contribution of this paper explicit, we emphasize three points. First, we apply the adaptive moments method (ADAM) to the transformed LHU objective defined on the nonsingular domain Ω . Second, we construct a mini-batch stochastic adaptive moments method (SADAM) for

large-scale LHU by using projected abundance variables within each batch. Third, for SADAM, we provide a conditional convergence analysis to stationary points under standard smoothness, boundedness, and tracking assumptions.

Model (1.3) has two variables and can be solved by an alternating minimization strategy. For fixed Q , the S -subproblem reduces to Euclidean projection onto the probability simplex $\mathcal{S}_p$ [3, 4],

$$\min_S \frac{1}{2} \|S - QY\|_F^2, \\ S \geq \mathbf{0}_{p \times n}, \mathbf{1}_p^\top S = \mathbf{1}_n^\top.$$

The S -subproblem has the closed-form solution $\hat{S}(Q) := \text{Proj}_{\mathcal{S}_p}(QY)$, where the projection is applied columnwise and $\text{Proj}_{\mathcal{S}_p}$ denotes the probability-simplex projector. Therefore, model (1.3) is equivalent to solving the reduced problem in Q only:

$$\min_Q \frac{1}{2} \|QY - \hat{S}\|_F^2 - \lambda \log |\det(Q)|. \quad (1.4)$$

Define

$$\text{dist}(z, \mathcal{S}_p) = \|z - \Pi_{\mathcal{S}_p}(z)\|_2,$$

where $\Pi_{\mathcal{S}_p}$ denotes the Euclidean projection onto $\mathcal{S}_p$. Then model (1.4) can be rewritten as

$$\psi(Q) = \frac{1}{2} \sum_{i=1}^n \text{dist}^2(Qy_i, \mathcal{S}_p) - \lambda \log |\det(Q)|.$$

Since the squared distance to a nonempty closed convex set is continuously differentiable and

$$\nabla_z \frac{1}{2} \text{dist}^2(z, \mathcal{S}_p) = z - \Pi_{\mathcal{S}_p}(z),$$

the gradient of model (1.4) is

$$\nabla \psi(Q) = (QY - \hat{S}(Q)) Y^\top - \lambda Q^{-T}.$$

Therefore, model (1.4) is continuously differentiable on Ω and can be solved by first-order methods. The L -smoothness required in the convergence analysis is imposed only on the region visited by the iterates through Assumption 4.1. ADAM can automatically adjust the step size and correct the iterative direction by using first-order and second-order moment information [5, 6]. In this paper, we choose ADAM to solve model (1.4). For large-scale scenes, the computational time and complexity become prohibitive, since evaluating the gradient of ADAM is expensive. Therefore, we introduce the stochastic method to reduce dimensionality [7]. We use the stochastic gradient instead of full gradient to construct iterative direction and establish SADAM for LHU. The numerical experiments demonstrate that the computational cost is reduced, and the accuracy is satisfactory.

The rest of the paper is structured as follows. Section 2 is devoted to ADAM for LHU. SADAM is given in Section 3. Section 4 provides a conditional convergence analysis of SADAM. Section 5 shows some numerical experiments. Section 6 concludes the paper and points out some future research directions.

Some notations used throughout this paper are as follows: We write $\mathbf{E}_{n \times n}$ for the $n \times n$ matrix whose entries are all equal to one. $\mathbf{1}_p$ and $\mathbf{1}_n$ stand for the all-ones vectors in $\mathbb{R}^p$ and $\mathbb{R}^n$, respectively. $\|A\|_F = \sqrt{\text{Tr}(AA^T)} = \sqrt{\sum_{i,j} a_{ij}^2}$, $\forall A \in \mathbb{R}^{m \times n}$. $\langle A, B \rangle = \text{Tr}(AB^T) = \sum_{i=1}^m \sum_{j=1}^n a_{ij}b_{ij}$, $\forall A, B \in \mathbb{R}^{m \times n}$. $I(A)$ denotes the indicator function of set A ; and $X \geq \mathbf{0}$ means $X_{ij} \geq 0$ for all entries (i, j) .

2. The adaptive moments method

ADAM can automatically adjust the step size and correct the iterative direction using the useful information in the first-order moment and second-order moment. At iteration k , we first compute the projected abundance matrix $\hat{S}^k := \hat{S}(Q^k) = \text{Proj}_{S_p}(Q^k Y)$ and then evaluate the gradient of model (1.4). Therefore, model (1.4) can be solved by ADAM. The specific update rules of ADAM are as follows:

$$g^k = (Q^k Y - \hat{S}^k) Y^T - \lambda (Q^k)^{-T}, \quad (2.1)$$

$$m^k = \rho_1 m^{k-1} + (1 - \rho_1) g^k, \quad (2.2)$$

$$\hat{m}^k = \frac{m^k}{1 - \rho_1^k}, \quad (2.3)$$

$$v^k = \rho_2 v^{k-1} + (1 - \rho_2) g^k \odot g^k, \quad (2.4)$$

$$\hat{v}^k = \frac{v^k}{1 - \rho_2^k}, \quad (2.5)$$

$$Q^{k+1} = Q^k - \alpha_k \frac{\hat{m}^k}{\sqrt{\hat{v}^k + \delta}}, \quad (2.6)$$

where $\odot$ is the component-wise multiplication of two matrices. The equation $\hat{v}^k + \delta$ denotes that the constant δ is added to every element of $\hat{v}^k$. The equation $\sqrt{\hat{v}^k + \delta}$ is the component-wise square root of matrix $\hat{v}^k + \delta$. The equation $\frac{\hat{m}^k}{\sqrt{\hat{v}^k + \delta}}$ denotes the component-wise division between $\hat{m}^k$, and $\sqrt{\hat{v}^k + \delta}$. m^k is the first-order moment, which is the average of the past gradient and the current gradient. v^k is the second moment. Moreover, the constant parameters ρ_1 and ρ_2 represent the decay rate of the first moment and the second moment, respectively. The step size α_k adopts a decaying strategy. To ensure the convergence of the algorithm, we utilize the inverse-time decay step size in the following form:

$$\alpha_k = \frac{\alpha^0}{(k+1)^\gamma}, \quad (2.7)$$

where $\alpha^0 > 0$ is the initial step size and $\gamma \in (0, 1]$ is the decay exponent. This step size strategy satisfies the theoretical requirements for step sizes in stochastic optimization algorithms, helping the algorithm to better balance exploration and convergence speed while ensuring convergence.

The ADAM for LHU is given in Algorithm 1.

Algorithm 1. Adaptive moments method (ADAM).

Input: Q^0 , $\alpha^0 > 0$, $\lambda > 0$, ρ_1 , ρ_2 , $\delta > 0$

- 1 Initialize $m^0 = 0$, $v^0 = 0$, and $k = 0$;
- 2 **while** *stopping criterion not satisfied* **do**
- 3 Compute $\hat{S}^k = \text{Proj}_{S_p}(Q^k Y)$ and then evaluate g^k by (2.1);
- 4 $m^{k+1} = \rho_1 m^k + (1 - \rho_1)g^k$, $v^{k+1} = \rho_2 v^k + (1 - \rho_2)g^k \odot g^k$;
- 5 Correct bias: $\hat{m}^{k+1} = \frac{m^{k+1}}{1 - \rho_1^{k+1}}$, $\hat{v}^{k+1} = \frac{v^{k+1}}{1 - \rho_2^{k+1}}$;
- 6 $Q^{k+1} = Q^k - \alpha_k \frac{\hat{m}^{k+1}}{\sqrt{\hat{v}^{k+1} + \delta}}$;
- 7 $k \leftarrow k + 1$;
- 8 **end**

3. The stochastic adaptive moments method (SADAM)

For each sample y_i , let

$$s_i(Q) = \Pi_{S_p}(Q y_i).$$

Then model (1.4) can be written in the normalized form

$$\min_Q \varphi(Q) = \frac{1}{2n} \sum_{i=1}^n \|Q y_i - s_i(Q)\|_2^2 - \frac{\lambda}{n} \log |\det(Q)|.$$

For convenience, we normalize the original objective

$$\psi(Q) = \frac{1}{2} \sum_{i=1}^n \|Q y_i - s_i(Q)\|_2^2 - \lambda \log |\det(Q)|$$

by the positive constant n and define $\varphi(Q) = \psi(Q)/n$. Thus, the factor $1/n$ is not an additional scaling applied only to the log-determinant term; rather, it appears in both the data-fidelity term and the log-determinant regularizer because the entire original objective is divided by n . Since multiplication by a positive constant does not change the stationary points, this normalization does not affect the stationary points of model (1.4). The gradient of $\varphi(Q)$ is

$$\nabla \varphi(Q) = \frac{1}{n} \sum_{i=1}^n \left[(Q y_i - s_i(Q)) y_i^\top \right] - \frac{\lambda}{n} Q^{-T}, \quad (3.1)$$

where n represents the number of mixed pixels. As can be seen from Eq (3.1), we must compute all gradients $\nabla \varphi_i(Q)$ ($i = 1, \dots, n$) in order to obtain $\nabla \varphi(Q)$ in every iteration. When the number of mixed pixels is large, computing $\nabla \varphi(Q)$ becomes expensive, and the computational cost is high.

In order to overcome this difficulty, we apply SADAM to solve model (1.4) in the large-scale setting. SADAM requires the use of only one (or a small set of) randomly selected sample(s) to form the stochastic gradient instead of the full gradient in each iteration [7]. Specifically, a subset of mixed pixels is randomly selected in each iteration, and its index set is denoted by B_k . Let Y_{B_k} be the corresponding sub-matrix, and define

$$\hat{S}_{B_k}^k := \left[s_i(Q^k) \right]_{i \in B_k} = \Pi_{S_p}(Q^k Y_{B_k}),$$

where the projection is taken columnwise. Then the stochastic gradient can be calculated as follows:

$$\begin{aligned} g^k &= \frac{1}{|B_k|} \sum_{i \in B_k} \left[(Q^k y_i - s_i(Q^k)) y_i^\top \right] - \frac{\lambda}{n} (Q^k)^{-T} \\ &= \frac{1}{|B_k|} (Q^k Y_{B_k} - \hat{S}_{B_k}^k) Y_{B_k}^\top - \frac{\lambda}{n} (Q^k)^{-T}. \end{aligned} \quad (3.2)$$

The proposed SADAM for LHU is summarized in Algorithm 2.

Algorithm 2. Stochastic adaptive moments method (SADAM).

Input: $Q^0, \tau^0 > 0, \lambda > 0, \rho_1, \rho_2, \delta > 0$
1 Initialize $H^0 = 0, G^0 = 0$, and $k = 0$;
2 *while* stopping criterion not satisfied **do**
3 Sample a mini-batch B_k and form Y_{B_k} ;
4 Compute $\hat{S}_{B_k}^k = \Pi_{S_p}(Q^k Y_{B_k})$ and then evaluate g^k by (3.2);
5 $H^{k+1} = \rho_1 H^k + (1 - \rho_1) g^k, G^{k+1} = \rho_2 G^k + (1 - \rho_2) g^k \odot g^k$;
6 Correct bias: $\hat{H}^{k+1} = \frac{H^{k+1}}{1 - \rho_1^{k+1}}, \hat{G}^{k+1} = \frac{G^{k+1}}{1 - \rho_2^{k+1}}$;
7 $Q^{k+1} = Q^k - \tau^k \frac{\hat{H}^{k+1}}{\sqrt{\hat{G}^{k+1} + \delta}}$;
8 $k \leftarrow k + 1$;
9 **end**

4. Convergence analysis of SADAM

This section analyzes SADAM. Since multiplying the objective by a positive constant does not change its stationary points, we use the normalized objective

$$\varphi(Q) = \frac{1}{2n} \sum_{i=1}^n \text{dist}^2(Q y_i, \mathcal{S}_p) - \frac{\lambda}{n} \log |\det(Q)|, \quad (4.1)$$

where

$$\mathcal{S}_p = \{s \in \mathbb{R}^p : s \geq 0, \mathbf{1}^\top s = 1\}, \quad \text{dist}(z, \mathcal{S}_p) = \|z - \Pi_{\mathcal{S}_p}(z)\|_2.$$

Here, $\Pi_{\mathcal{S}_p}$ denotes the Euclidean projection onto $\mathcal{S}_p$. For each sample, define

$$s_i(Q) = \Pi_{\mathcal{S}_p}(Q y_i), \quad r_i(Q) = Q y_i - s_i(Q).$$

Then

$$\nabla \varphi(Q) = \frac{1}{n} \sum_{i=1}^n r_i(Q) y_i^\top - \frac{\lambda}{n} Q^{-T}. \quad (4.2)$$

Consequently, if $B_k \subset \{1, \dots, n\}$ is sampled uniformly, the mini-batch gradient estimator is

$$g^k = \frac{1}{|B_k|} \sum_{i \in B_k} r_i(Q^k) y_i^\top - \frac{\lambda}{n} (Q^k)^{-T}. \quad (4.3)$$

The use of Q^{-T} follows from the identity $\nabla_Q \log |\det(Q)| = Q^{-T}$ under the Frobenius inner product.

Let $\mathcal{F}_k$ be the sigma-algebra generated by the history before drawing B_k . The SADAM direction is written as

$$d^k = \frac{\hat{H}^k}{\sqrt{\hat{G}^k + \delta}}, \quad (4.4)$$

where all operations in the quotient are component-wise, and the update is

$$Q^{k+1} = Q^k - \tau_k d^k. \quad (4.5)$$

Assumption 4.1 (Smoothness and lower boundedness). *The iterates generated by SADAM are well defined and remain in the domain of φ . Moreover, φ is continuously differentiable and L -smooth; that is, there exists a constant $L > 0$ such that*

$$\|\nabla\varphi(Q) - \nabla\varphi(\tilde{Q})\|_F \leq L\|Q - \tilde{Q}\|_F.$$

In addition, there exists $\varphi_{\inf} > -\infty$ such that $\varphi(Q^k) \geq \varphi_{\inf}$ for all sufficiently large k .

Remark 4.1. *Assumption 4.1 should be understood as a conditional assumption imposed on the region visited by the iterates. This point is important for the present LHU objective: The projection-distance term has Lipschitz continuous gradient, but the log-determinant term is differentiable only on the nonsingular domain $\Omega = \{Q : \det(Q) \neq 0\}$. Consequently, if Q approaches singularity, the gradient Q^{-T} may become unbounded, and a global L -smoothness property generally fails. Therefore, Assumption 4.1 is used only on a bounded nonsingular region containing the generated iterates or equivalently on a region visited by the iterates on which they remain uniformly away from singularity. This is exactly the domain-stability interpretation needed in the subsequent descent analysis.*

At iteration k , the mini-batch B_k is sampled uniformly from $\{1, \dots, n\}$ and independently of $\mathcal{F}_k$, with $|B_k| \geq 1$. Therefore, the mini-batch estimator (4.3) is unbiased:

$$\mathbb{E}[g^k \mid \mathcal{F}_k] = \nabla\varphi(Q^k).$$

This follows directly from the uniform sampling rule. To control the stochastic error, we further impose the following variance and boundedness assumption.

Assumption 4.2 (Variance and bounded stochastic gradients). *There is a nonnegative sequence $\{\sigma_k^2\}$ such that*

$$\mathbb{E}[\|g^k - \nabla\varphi(Q^k)\|_F^2 \mid \mathcal{F}_k] \leq \sigma_k^2.$$

In addition, the stochastic gradients are uniformly bounded along the generated sequence; i.e., there exists $G > 0$ such that

$$\|g^k\|_F \leq G$$

almost surely for all sufficiently large k . Consequently, the conditional second moment is bounded, namely there exists $M_g > 0$ such that

$$\mathbb{E}[\|g^k\|_F^2 \mid \mathcal{F}_k] \leq M_g^2.$$

Assumption 4.3 (Step sizes). *The step sizes satisfy*

$$\tau_k > 0, \quad \sum_{k=0}^{\infty} \tau_k = \infty, \quad \sum_{k=0}^{\infty} \tau_k^2 < \infty.$$

In particular, $\tau_k = \tau_0/(k+1)^\gamma$ satisfies this assumption whenever $\tau_0 > 0$ and $1/2 < \gamma \leq 1$.

Assumption 4.4 (Summable tracking error). *The tracking error sequence*

$$\eta_k := \mathbb{E} \|\hat{H}^k - \nabla \varphi(Q^k)\|_F^2$$

satisfies

$$\sum_{k=0}^{\infty} \tau_k \eta_k < \infty. \quad (4.6)$$

Lemma 4.1 (Gradient properties). *Under Assumptions 4.1 and 4.2, the following hold.*

- (1) *The function φ is continuously differentiable, and φ is L -smooth in the sense of Assumption 4.1.*
- (2) *There exists $G > 0$ such that $\|g^k\|_F \leq G$ almost surely for all sufficiently large k . Consequently, the denominator in (4.4) is component-wise bounded as*

$$\sqrt{\delta} \leq \sqrt{\hat{G}_{ab}^k + \delta} \leq \sqrt{G^2 + \delta} \quad \text{for all coordinates } (a, b), \quad (4.7)$$

up to harmless finite initial transients.

Proof. The squared distance to a nonempty closed convex set is continuously differentiable and

$$\nabla_z \frac{1}{2} \text{dist}^2(z, \mathcal{S}_p) = z - \Pi_{\mathcal{S}_p}(z).$$

Together with the differentiability of the log-determinant term on the nonsingular domain, this gives the gradient formula (4.2). The L -smoothness of φ is imposed directly in Assumption 4.1. The almost sure boundedness of g^k follows from Assumption 4.2. The component-wise upper bound in (4.7) follows because $\hat{G}^k$ is a bias-corrected exponentially weighted average of $g^t \odot g^t$, while the lower bound follows from $\delta > 0$. $\square$

The next lemma explains why the first moment $\hat{H}^k$ can be regarded as an approximate current gradient when the iterates move slowly and the stochastic gradient noise is controlled. This is the point at which the analysis uses the ADAM-type momentum structure.

Lemma 4.2 (Moment tracking). *Under Assumptions 4.1 and 4.2 and using the uniform sampling rule stated above, suppose that the bias-corrected first moment can be written as*

$$\hat{H}^k = \sum_{t=0}^k a_{k,t} g^t, \quad a_{k,t} \geq 0, \quad \sum_{t=0}^k a_{k,t} = 1,$$

with exponentially decaying weights generated by the parameter $\rho_1 \in (0, 1)$. More precisely, for $k \geq 1$,

$$a_{k,t} = \begin{cases} \frac{(1 - \rho_1)\rho_1^{k-1-t}}{1 - \rho_1^k}, & t = 0, \dots, k-1, \\ 0, & t = k, \end{cases}$$

which is the explicit weight formula obtained from the bias-corrected Adam recursion with $H^0 = 0$. Let

$$\xi^t = g^t - \nabla\varphi(Q^t).$$

Then

$$\mathbb{E}\|\hat{H}^k - \nabla\varphi(Q^k)\|_F^2 \leq 2L^2\mathbb{E}\left(\sum_{t=0}^k a_{k,t} \sum_{j=t}^{k-1} \tau_j \|d^j\|_F\right)^2 + 2 \sum_{t=0}^k a_{k,t}^2 \mathbb{E}[\sigma_t^2]. \quad (4.8)$$

In particular, if $\sup_j \|d^j\|_F \leq D$ and $\tau_k = \tau_0/(k+1)^\gamma$ with $\gamma \in (0, 1]$, then, for all sufficiently large k ,

$$\mathbb{E}\|\hat{H}^k - \nabla\varphi(Q^k)\|_F^2 \leq C_H \tau_k^2 + C_H \sum_{t=0}^k a_{k,t}^2 \mathbb{E}[\sigma_t^2], \quad (4.9)$$

where $C_H > 0$ is independent of k .

Proof. From the recursion

$$H^k = (1 - \rho_1) \sum_{t=0}^{k-1} \rho_1^{k-1-t} g^t$$

and the bias correction $\hat{H}^k = H^k/(1 - \rho_1^k)$, we obtain the stated explicit weights $a_{k,t}$. In particular, $a_{k,t} \geq 0$ and $\sum_{t=0}^k a_{k,t} = 1$. Decompose

$$\hat{H}^k - \nabla\varphi(Q^k) = \sum_{t=0}^k a_{k,t} (\nabla\varphi(Q^t) - \nabla\varphi(Q^k)) + \sum_{t=0}^k a_{k,t} \xi^t.$$

By Lipschitz continuity and the update rule,

$$\|\nabla\varphi(Q^t) - \nabla\varphi(Q^k)\|_F \leq L \sum_{j=t}^{k-1} \|Q^{j+1} - Q^j\|_F = L \sum_{j=t}^{k-1} \tau_j \|d^j\|_F.$$

The unbiasedness relation stated above gives the martingale-difference property $\mathbb{E}[\xi^t \mid \mathcal{F}_t] = 0$, which yields

$$\mathbb{E}\left\|\sum_{t=0}^k a_{k,t} \xi^t\right\|_F^2 = \sum_{t=0}^k a_{k,t}^2 \mathbb{E}\|\xi^t\|_F^2 \leq \sum_{t=0}^k a_{k,t}^2 \mathbb{E}[\sigma_t^2].$$

Using $(a+b)^2 \leq 2a^2 + 2b^2$ gives (4.8). For exponentially decaying weights and polynomially decaying step sizes, the weighted history length is bounded by a constant depending only on ρ_1 , so

$$\sum_{t=0}^k a_{k,t} \sum_{j=t}^{k-1} \tau_j \leq C_{\rho_1} \tau_k$$

for all large k . This proves (4.9). $\square$

Remark 4.2 (A sufficient condition for Assumption 4.4). *Lemma 4.2 provides one way to verify Assumption 4.4. In particular, if $\sup_j \|d^j\|_F \leq D$, the step sizes have the form $\tau_k = \tau_0/(k+1)^\gamma$ with $1/2 < \gamma \leq 1$, and*

$$\sum_{k=0}^{\infty} \tau_k \sum_{t=0}^k a_{k,t}^2 \mathbb{E}[\sigma_t^2] < \infty,$$

then (4.9) implies (4.6). This can occur, for example, when the mini-batch variance is reduced along the iterations.

The following proposition follows the gradient-related direction argument used in ADAM-type convergence analyses [5, 8], adapted to the SADAM direction for the LHU objective.

Proposition 4.1 (Gradient-related SADAM direction). *Suppose Assumptions 4.1–4.4 hold. Then there exist constants $\mu > 0$, $C_0 \geq 0$ and $C_1 \geq 0$, and a nonnegative sequence $\{\varepsilon_k\}$ satisfying*

$$\sum_{k=0}^{\infty} \tau_k \varepsilon_k < \infty$$

such that, for all sufficiently large k ,

$$\mathbb{E}[\langle \nabla \varphi(Q^k), d^k \rangle] \geq \mu \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 - \varepsilon_k, \quad (4.10)$$

$$\mathbb{E}[\|d^k\|_F^2] \leq C_0 + C_1 \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 + \varepsilon_k. \quad (4.11)$$

Proof. By Lemma 4.1, the denominator of the adaptive direction is bounded component-wise. Let

$$\underline{v} = \sqrt{\delta}, \quad \bar{v} = \sqrt{G^2 + \delta}.$$

For any matrix A , define the component-wise preconditioner

$$(P_k \odot A)_{ab} = \frac{A_{ab}}{\sqrt{\hat{G}_{ab}^k + \delta}}.$$

Then $d^k = P_k \odot \hat{H}^k$ and

$$\langle \nabla \varphi(Q^k), P_k \odot \nabla \varphi(Q^k) \rangle \geq \frac{1}{\bar{v}} \|\nabla \varphi(Q^k)\|_F^2.$$

Moreover,

$$|\langle \nabla \varphi(Q^k), P_k \odot (\hat{H}^k - \nabla \varphi(Q^k)) \rangle| \leq \frac{1}{\underline{v}} \|\nabla \varphi(Q^k)\|_F \|\hat{H}^k - \nabla \varphi(Q^k)\|_F.$$

Young's inequality with $a = 1/(2\bar{v})$ gives

$$\mathbb{E} \langle \nabla \varphi(Q^k), d^k \rangle \geq \frac{1}{2\bar{v}} \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 - \frac{\bar{v}}{2\underline{v}^2} \eta_k.$$

Similarly,

$$\|d^k\|_F^2 \leq \frac{1}{\underline{v}^2} \|\hat{H}^k\|_F^2 \leq \frac{2}{\underline{v}^2} \|\nabla \varphi(Q^k)\|_F^2 + \frac{2}{\underline{v}^2} \|\hat{H}^k - \nabla \varphi(Q^k)\|_F^2.$$

Taking expectation proves (4.10)-(4.11) with ε_k proportional to η_k . $\square$

The proof of the next theorem follows the standard descent-summability framework for stochastic optimization [9], after applying the gradient-related property established in Proposition 4.1.

Theorem 4.1 (Convergence to stationary points). *Suppose Assumptions 4.1–4.4 hold. Then the SADAM iterates satisfy*

$$\sum_{k=0}^{\infty} \tau_k \mathbb{E} \left[\|\nabla \varphi(Q^k)\|_F^2 \right] < \infty. \quad (4.12)$$

Consequently,

$$\liminf_{k \rightarrow \infty} \mathbb{E} \left[\|\nabla \varphi(Q^k)\|_F^2 \right] = 0. \quad (4.13)$$

Furthermore, if the bounds in Proposition 4.1 hold pathwise and $\sup_k \|d^k\|_F < \infty$ almost surely, then

$$\|\nabla \varphi(Q^k)\|_F \rightarrow 0 \quad \text{a.s.} \quad (4.14)$$

Proof. By Lemma 4.1, φ is L -smooth. Using (4.5), we obtain

$$\varphi(Q^{k+1}) \leq \varphi(Q^k) - \tau_k \langle \nabla \varphi(Q^k), d^k \rangle + \frac{L}{2} \tau_k^2 \|d^k\|_F^2.$$

Taking expectation and applying Proposition 4.1 gives

$$\begin{aligned} \mathbb{E}[\varphi(Q^{k+1})] &\leq \mathbb{E}[\varphi(Q^k)] - \mu \tau_k \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 + \tau_k \varepsilon_k \\ &\quad + \frac{L}{2} \tau_k^2 (C_0 + C_1 \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 + \varepsilon_k). \end{aligned}$$

Since $\tau_k \rightarrow 0$, there exists k_0 such that $(LC_1/2)\tau_k \leq \mu/2$ for all $k \geq k_0$. Hence, for all $k \geq k_0$,

$$\mathbb{E}[\varphi(Q^{k+1})] \leq \mathbb{E}[\varphi(Q^k)] - \frac{\mu}{2} \tau_k \mathbb{E} \|\nabla \varphi(Q^k)\|_F^2 + C \tau_k^2 + C \tau_k \varepsilon_k,$$

where $C > 0$ is independent of k . Summing from k_0 to N yields

$$\frac{\mu}{2} \sum_{k=k_0}^N \tau_k \mathbb{E} \left[\|\nabla \varphi(Q^k)\|_F^2 \right] \leq \mathbb{E}[\varphi(Q^{k_0})] - \mathbb{E}[\varphi(Q^{N+1})] + C \sum_{k=k_0}^N \tau_k^2 + C \sum_{k=k_0}^N \tau_k \varepsilon_k.$$

By Assumption 4.1, $\varphi(Q^k) \geq \varphi_{\inf} > -\infty$ along the generated sequence. Using Assumptions 4.3 and 4.4, the right-hand side is uniformly bounded as $N \rightarrow \infty$. This proves (4.12).

Since $\sum_k \tau_k = \infty$, (4.12) implies (4.13); otherwise, if the expected squared gradient were bounded below by a positive constant eventually, the series in (4.12) would diverge.

For the almost sure statement, the same descent inequality together with the Robbins–Siegmund supermartingale theorem [10] implies

$$\sum_{k=0}^{\infty} \tau_k \|\nabla \varphi(Q^k)\|_F^2 < \infty \quad \text{a.s.,}$$

if $\sup_k \|d^k\|_F \leq D$ almost surely, then

$$\|Q^{k+1} - Q^k\|_F \leq D \tau_k,$$

and Lipschitz continuity of $\nabla\varphi$ gives

$$\|\nabla\varphi(Q^{k+1}) - \nabla\varphi(Q^k)\|_F \leq LD\tau_k.$$

A standard contradiction argument now yields $\|\nabla\varphi(Q^k)\|_F \rightarrow 0$ almost surely: If the gradient norm exceeded some $\epsilon_0 > 0$ infinitely often, Lipschitz continuity and the step bound would keep it above $\epsilon_0/2$ over intervals with a fixed positive accumulated step size, contradicting the summability of $\sum_k \tau_k \|\nabla\varphi(Q^k)\|_F^2$. This proves (4.14). $\square$

Corollary 4.1 (Iteration complexity). *Suppose Assumptions 4.1–4.4 hold. Let $K \geq k_0$ and define*

$$S_K = \sum_{k=k_0}^K \tau_k.$$

Choose an index $R_K \in \{k_0, \dots, K\}$ randomly with probability

$$\mathbb{P}(R_K = k) = \frac{\tau_k}{S_K}.$$

Then

$$\mathbb{E}\|\nabla\varphi(Q^{R_K})\|_F^2 \leq \frac{2(\mathbb{E}\varphi(Q^{k_0}) - \varphi_{\inf}) + C \sum_{k=k_0}^K \tau_k^2 + C \sum_{k=k_0}^K \tau_k \varepsilon_k}{\mu S_K}. \quad (4.15)$$

Consequently, if $\sum_k \tau_k \varepsilon_k < \infty$ and $\tau_k = \tau_0/(k+1)^\gamma$ with $1/2 < \gamma < 1$, then

$$\mathbb{E}\|\nabla\varphi(Q^{R_K})\|_F^2 = O(K^{-(1-\gamma)}),$$

while for $\gamma = 1$ one obtains

$$\mathbb{E}\|\nabla\varphi(Q^{R_K})\|_F^2 = O(1/\log K).$$

Proof. The bound follows by dividing the finite-sum descent estimate in the proof of Theorem 4.1 by S_K and using the definition of R_K . $\square$

Remark 4.3 (Scope of the theory). *The preceding theorem should be interpreted as a conditional convergence theorem for SADAM under smoothness, boundedness, and moment-tracking conditions. This is intentional: For ADAM-type methods, the main theoretical difficulty is that the adaptive direction $\hat{H}^k / \sqrt{\hat{G}^k + \delta}$ is not an unbiased gradient and need not be gradient-related automatically. Similar issues and remedies, including Adaptive Mean Square Gradient (AMSGrad)-type denominators and diminishing momentum parameters, have been discussed in the Adam convergence literature [5, 8, 9]. Thus, the present analysis does not claim to solve the general Adam convergence problem; rather, it identifies verifiable stability and tracking conditions under which SADAM converges to stationary points of the LHU objective.*

5. Numerical experiments

This section evaluates the two proposed solvers, ADAM and SADAM (Algorithms 1 and 2), on the four benchmark hyperspectral scenes displayed in Figure 1. The scenes considered are Samson, Jasper Ridge, Urban and Cuprite [11–13]. Table 1 reports the spatial size of each image together with the number of endmembers it contains. All four scenes are publicly available at http://www.escience.cn/people/feiyunZHU/Dataset_GT.html.

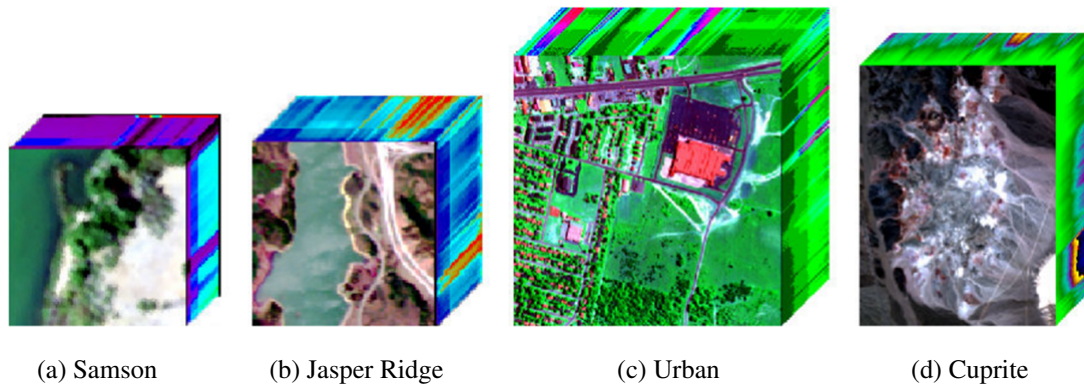


Figure 1. The four hyperspectral scenes used in the experiments: Samson, Jasper Ridge, Urban and Cuprite (from left to right).

Table 1. Main characteristics of the test scenes. Image size is given as (number of spectral bands) $\times$ (number of pixels).

Data set	Size	Number of endmembers(p)
Samson	156×9025	3
Jasper Ridge	198×10000	4
Urban	162×94249	6
Cuprite	188×47500	12

The performance of ADAM and SADAM is assessed against four reference unmixing methods:

- Simplex identification via split augmented Lagrangian (SISAL). We use the implementation available at <http://www.lx.it.pt/biucas/code.htm>.
- Mini-batch stochastic gradient descent (MBSGD) [14]. As no public implementation is provided, we code this method ourselves.
- Online kernel nonnegative matrix factorization (OKNMF) [15]. The implementation released by the authors is available at <http://www.honeine.fr/paul/publi/17.oknmf.zip>.
- SPGM [16]. It is included as an additional baseline.

Each method is coded in MATLAB R2017a. The experiments are carried out on a desktop computer equipped with an Intel Core i7-11370 processor (3.30 GHz) and 16 GB of RAM, under the Windows 10 operating system.

Unmixing accuracy is measured by the spectral angle distance (SAD) [17], which is the angle between an estimated endmember and its corresponding ground truth and is widely used to assess the quality of estimated endmembers. It is defined by

$$\text{SAD}(\mathbf{M}_i, \hat{\mathbf{M}}_i) = \arccos\left(\frac{\mathbf{M}_i^T \hat{\mathbf{M}}_i}{\|\mathbf{M}_i\| \cdot \|\hat{\mathbf{M}}_i\|}\right),$$

Here M_i denotes the reference spectrum of the i th endmember, whereas $\hat{M}_i$ is the estimate produced by an unmixing algorithm. For LHU, a smaller SAD value indicates better unmixing performance.

To improve reproducibility, we summarize the main implementation settings used in our experiments. SADAM, ADAM, SPGM, SISAL and MBSGD are allowed at most 2500 iterations, whereas OKNMF is limited to 10 iterations [18]. For ADAM and SADAM, we set $\rho_1 = 0.82$ and $\rho_2 = 0.99$. These values are selected by preliminary tuning around the standard Adam choices and are then kept fixed for all data sets. We use the decaying step-size rules in (2.7) and Assumption 4.3. The initial matrix Q^0 is taken as the identity matrix in the reduced p -dimensional space, and the simplex projection is computed by the Euclidean projection onto $\mathcal{S}_p$ applied columnwise. For SADAM, the mini-batch size is fixed throughout each run. The regularization parameter λ is chosen from a small validation grid for each data set and then kept fixed during the corresponding experiment. The reported stochastic results are obtained from one representative stochastic run. Default parameter values are kept for SISAL, MBSGD and OKNMF. SPGM [16] is executed under the parameter setting adopted in this work, namely the decaying step-size rule (2.7) and the initial point $Q^0 = I_p$.

Tables 2–5 collect the SAD values and the CPU time of the six methods. The scenes differ in the number of endmembers: Samson, for instance, contains the three endmembers rock, tree and water. In Table 2, the mean value is the average SAD over the three endmembers, and Tables 3–5 follow the same notation as Table 2.

As can be seen from Tables 2–5, SADAM attains the smallest mean SAD on the Samson, Urban and Cuprite scenes. On the Jasper Ridge scene, SPGM gives a slightly smaller mean SAD than SADAM, namely 0.3123 against 0.3243. In terms of CPU time, SPGM is the fastest of the six methods on the Samson, Jasper Ridge and Urban scenes. On the Cuprite scene, SADAM and SPGM require the same CPU time and are clearly faster than the remaining methods.

A further remark follows from these results. The CPU-time gap between SPGM and SADAM is much smaller on the two large scenes than on the two small ones: SPGM is about three times faster on Samson and about ten times faster on Jasper Ridge, but only about 1.6 times faster on Urban, and the two methods take the same time (0.2 s) on Cuprite. Note that these runtimes are not determined by the scene size alone: SADAM is faster on the larger Cuprite scene (0.2 s) than on the smaller Jasper Ridge one (0.5 s). This is consistent with the different character of the two methods, SADAM being an adaptive-moment method with a decaying step size whereas SPGM is a proximal-gradient method.

Table 2. Unmixing accuracy (SAD) and CPU time on the Samson scene.

Algorithms	SAD				Time (s)
	1-rock	2-tree	3-water	Mean value	
SADAM	0.0116	0.0509	0.1766	0.0797	0.2372
SPGM	0.0519	0.0142	0.1767	0.0810	0.0782
ADAM	0.0702	0.0805	0.2222	0.1243	4.3804
SISAL	0.2103	0.1023	0.2136	0.1754	5.2494
MBSGD	0.1042	0.0275	0.2916	0.1411	21.9226
OKNMF	0.0959	0.3640	0.7639	0.4080	38.3780

Table 3. Unmixing accuracy (SAD) and CPU time on the Jasper Ridge scene.

Algorithms	SAD					Time (s)
	1-tree	2-water	3-dirt	4-road	Mean value	
SADAM	0.1826	0.2106	0.2735	0.6305	0.3243	0.5002
SPGM	0.1220	0.0396	0.1349	0.9533	0.3123	0.0493
ADAM	0.2626	0.2733	0.4516	0.6820	0.4173	1.7670
SISAL	0.2917	0.3295	0.5657	0.7558	0.4857	4.2991
MBSGD	0.1562	0.2417	0.2241	0.6873	0.3273	23.1698
OKNMF	0.2907	0.6039	0.3857	0.3619	0.4105	56.3967

Table 4. Unmixing accuracy (SAD) and CPU time on the Urban scene.

Algorithms	SAD						Time (s)
	1-asphalt	2-grass	3-tree	4-roof	5-metal	6-dirt	
SADAM	0.3030	0.1219	0.3357	0.4609	0.3622	0.3379	0.8000
SPGM	0.3115	0.2662	0.2128	0.5296	0.1024	1.0651	0.5000
ADAM	0.3896	0.1894	0.3071	0.5035	0.4645	0.4528	20.9000
SISAL	0.6295	0.2626	0.3470	0.5254	1.1615	1.3165	95.1000
MBSGD	0.7302	0.5378	0.4482	0.5816	0.6968	1.3228	190.0000
OKNMF	0.2257	0.3511	0.3015	0.2597	0.9027	1.1645	1257.5000

Table 5. Unmixing accuracy (SAD) and CPU time on the Cuprite scene.

SAD	Algorithms					
	SADAM	SPGM	ADAM	SISAL	MBSGD	OKNMF
1-Alunite	0.0964	0.1376	0.2179	0.2023	0.2420	0.9962
2-Andradite	0.0681	0.0743	0.0769	0.1077	0.2407	0.9045
3-Buddingtonite	0.0626	0.1104	0.1533	0.0975	0.3347	1.0519
4-Dumortierite	0.1184	0.1236	0.1452	0.1365	0.3208	1.0139
5-Kaolinite1	0.0683	0.0968	0.1473	0.1249	0.2820	1.0122
6-Kaolinite1	0.0762	0.0760	0.1446	0.1661	0.3948	1.0921
7-Muscovite	0.1910	0.1724	0.2033	0.2738	0.3175	1.2457
8-Montmorillonite	0.1163	0.0914	0.1627	0.2439	0.4060	1.2845
9-Nontronite	0.1462	0.1207	0.2013	0.2471	0.4522	1.3030
10-Pyrope	0.0776	0.2053	0.2465	0.3275	0.3141	1.3353
11-Sphene	0.0581	0.2475	0.3450	0.6287	0.6912	1.4491
12-Chalcedony	1.2512	1.0938	1.0719	1.3673	1.4159	1.4306
Mean Value	0.1942	0.2125	0.2597	0.3270	0.4510	1.1766
Time (s)	0.2000	0.2000	43.0000	101.2000	7.9000	1077.6000

Figures 2–4 compare the spectral signatures recovered by the six methods on Samson, Jasper Ridge and Urban. The Cuprite scene, which contains 12 endmembers, is omitted here for brevity. Overall, the spectral signatures produced by SADAM are close to the ground truth and are visually competitive with those of the other methods.

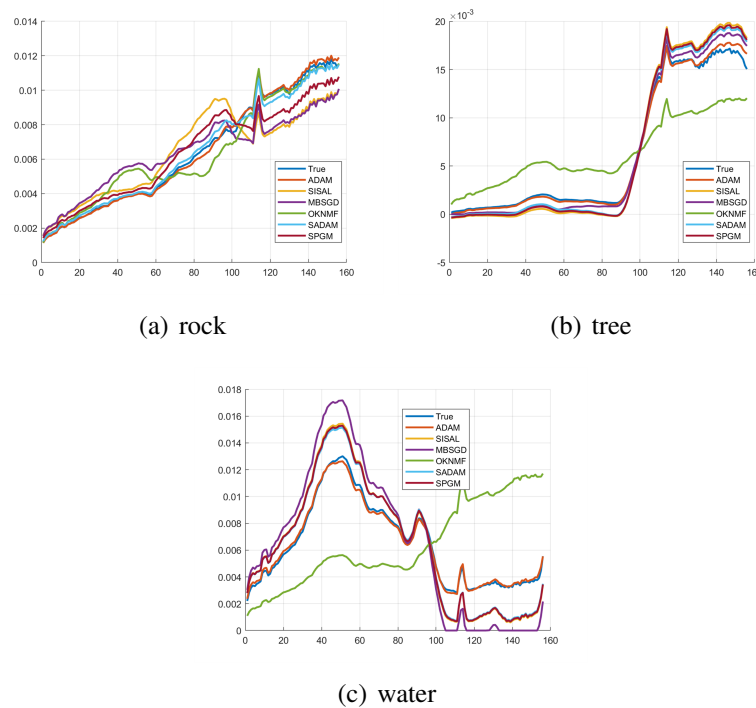


Figure 2. Estimated spectral signatures on Samson: comparison of the six methods with the ground truth.

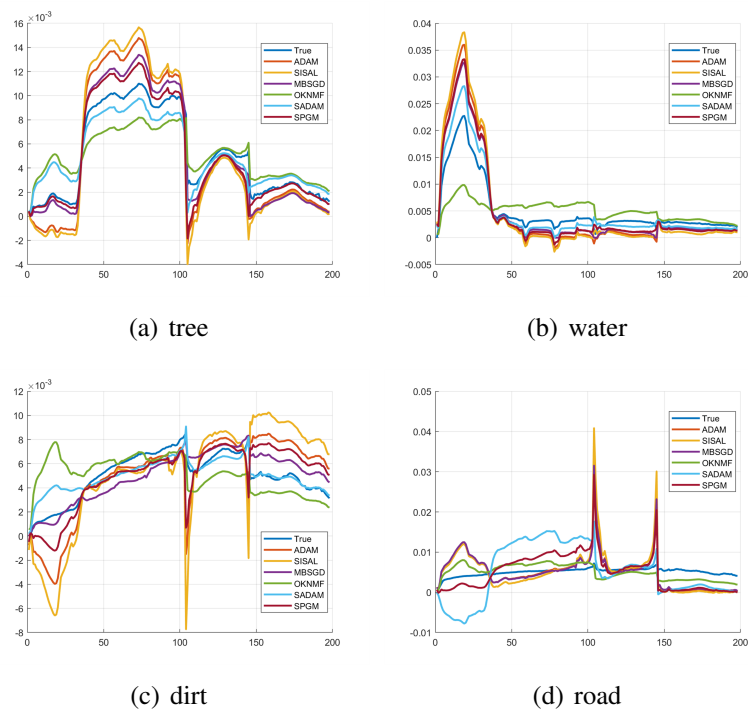


Figure 3. Estimated spectral signatures on Jasper Ridge: comparison of the six methods with the ground truth.

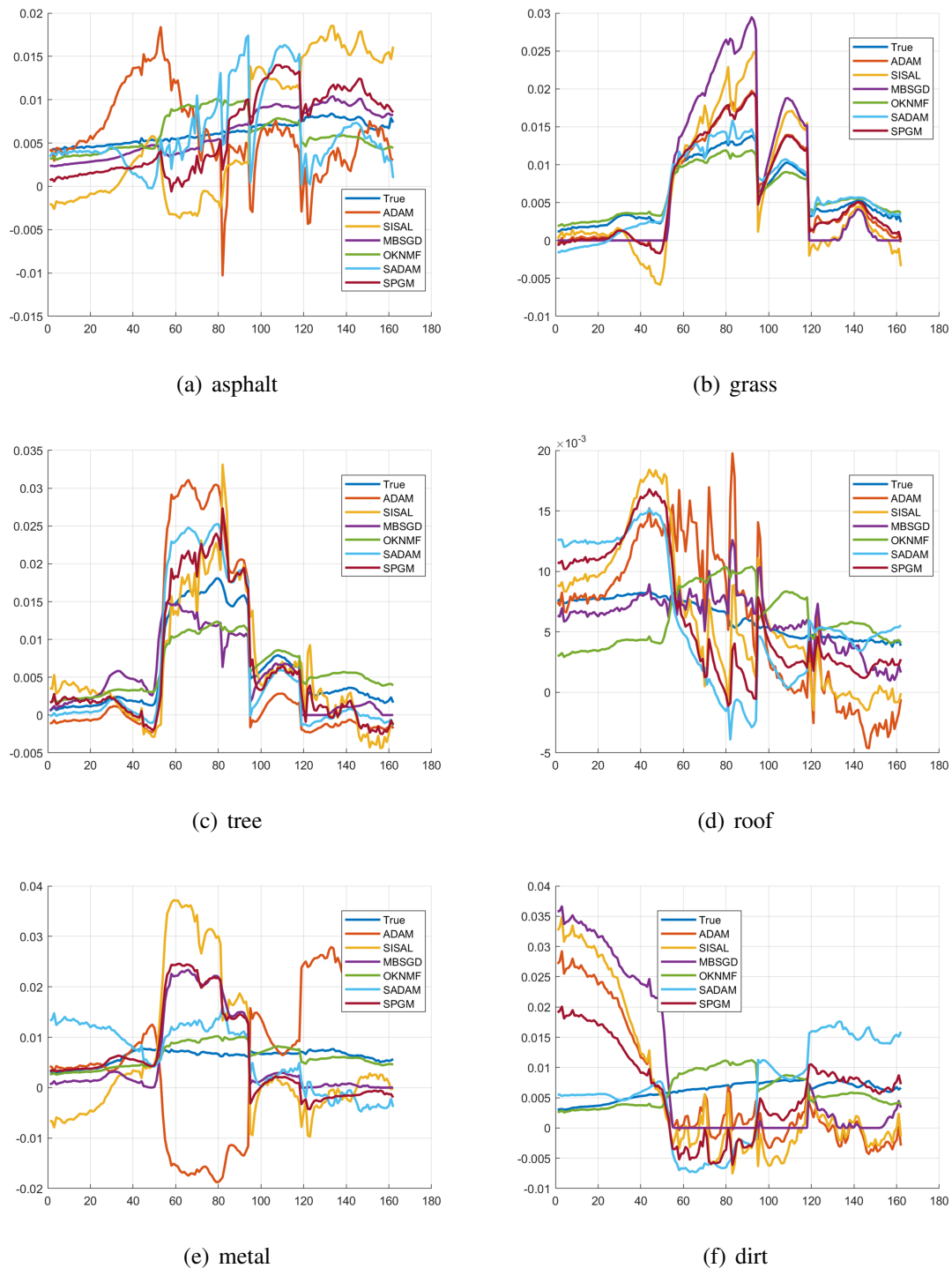
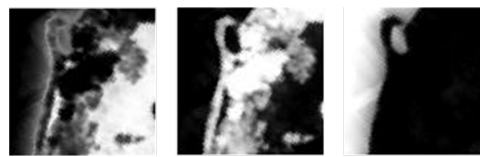


Figure 4. Estimated spectral signatures on Urban: comparison of the six methods with the ground truth.

Abundance maps obtained on Samson and Jasper Ridge are shown in Figures 5 and 6. The first panel of each figure gives the ground truth, and the remaining panels correspond to SADAM, SPGM,

ADAM, SISAL, MBSGD and OKNMF, respectively. SADAM yields abundance maps with clear spatial structures and visually reasonable agreement with the ground truth.



(a) Ground truth: Abundances



(b) Abundances of SADAM



(c) Abundances of SPGM



(d) Abundances of ADAM



(e) Abundances of SISAL

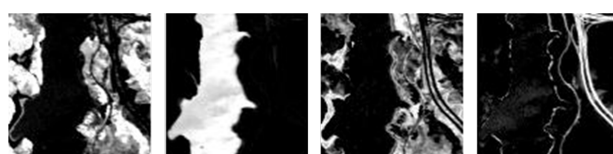


(f) Abundances of MBSGD

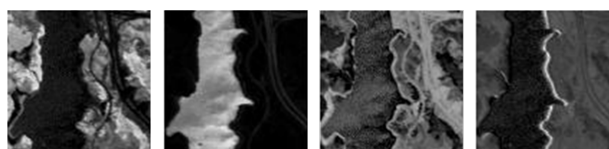


(g) Abundances of OKNMF

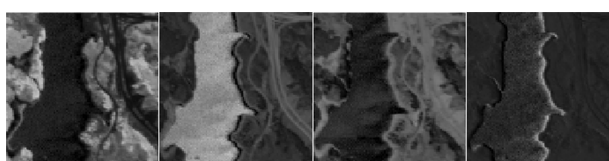
Figure 5. Abundance maps on Samson: comparison of the six methods with the ground truth.



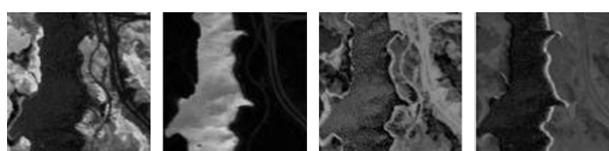
(a) Ground truth: Abundances



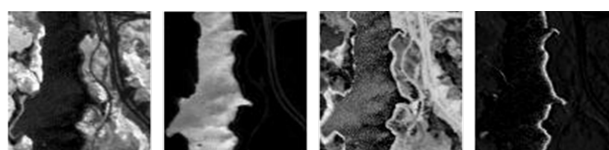
(b) Abundances of SADAM



(c) Abundances of SPGM



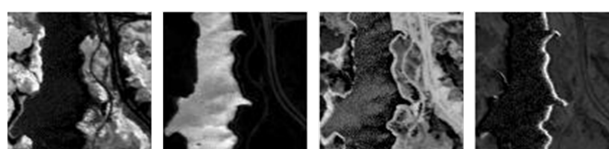
(d) Abundances of ADAM



(e) Abundances of SISAL



(f) Abundances of MBSGD



(g) Abundances of OKNMF

Figure 6. Abundance maps on Jasper Ridge: comparison of the six methods with the ground truth.

Taken together, the above results indicate that the proposed SADAM behaves well on the real data sets considered here.

6. Conclusions

In this paper, we study ADAMs for the transformed LHU model. By introducing the inverse variable $Q = M^{-1}$ after PCA-based dimensionality reduction, we obtain a smooth optimization model on the nonsingular domain and solve it by ADAM and the proposed SADAM. For SADAM, we derive a mini-batch gradient formula based on projected abundance variables and establish a conditional convergence result to stationary points under smoothness, boundedness, and moment-tracking assumptions. The experiments on four real hyperspectral data sets confirm that SADAM cuts the running time while remaining competitive in unmixing accuracy.

Author contributions

Zhewei Zhang: methodology, software and experiments, formal analysis, writing-original draft, and writing-review and editing. Fangfang Xu: conceptualization, methodology, software and experiments, formal analysis, writing-original draft, supervision, and funding acquisition. All authors have read and approved the final manuscript.

Use of Generative-AI tools declaration

During the preparation of this work, the authors used ChatGPT to polish the language and translate parts of the text. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Acknowledgments

This work was funded by the Natural Science Foundation of China (No. W2612006).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. F. Xu, Y. Wang, Y. Li, L. Liu, T. Tian, Gradient type methods for linear hyperspectral unmixing, *CSIAM Trans. Appl. Math.*, **3** (2022), 109–132. <https://doi.org/10.4208/csiam-am.SO-2021-0001>
2. R. Bro, A. K. Smilde, Principal component analysis, *Anal. Methods*, **6** (2014), 2812–2831. <https://doi.org/10.1039/c3ay41907j>
3. J. Duchi, S. Shalev-Shwartz, Y. Singer, T. Chandra, Efficient projections onto the ℓ_1 -ball for learning in high dimensions, *Proceedings of the 25th International Conference on Machine Learning*, 2008, 272–279. <https://doi.org/10.1145/1390156.1390191>

4. W. Wang, M. A. Carreira-Perpiñán, Projection onto the probability simplex: an efficient algorithm with a simple proof, and an application, *arXiv*, 2013. <https://doi.org/10.48550/arXiv.1309.1541>
5. X. Chen, S. Liu, R. Sun, M. Hong, On the convergence of a class of Adam-type algorithms for non-convex optimization, *arXiv*, 2018. <https://doi.org/10.48550/arXiv.1808.02941>
6. D. P. Kingma, J. Ba, Adam: a method for stochastic optimization, *arXiv*, 2014. <https://doi.org/10.48550/arXiv.1412.6980>
7. W. B. Powell, A unified framework for stochastic optimization, *Eur. J. Oper. Res.*, **275** (2019), 795–821. <https://doi.org/10.1016/j.ejor.2018.07.014>
8. S. J. Reddi, S. Kale, S. Kumar, On the convergence of Adam and beyond, *arXiv*, 2018. <https://doi.org/10.48550/arXiv.1904.09237>
9. X. Li, A. Milzarek, A unified convergence theorem for stochastic optimization methods, *arXiv*, 2022. <https://doi.org/10.48550/arXiv.2206.03907>
10. H. Robbins, D. Siegmund, A convergence theorem for non negative almost supermartingales and some applications, In: *Optimizing methods in statistics*, New York: Academic Press, 1971, 233–257. <https://doi.org/10.1016/B978-0-12-604550-5.50015-8>
11. F. Zhu, Y. Wang, B. Fan, G. Meng, C. Pan, Effective spectral unmixing via robust representation and learning-based sparsity, *arXiv*, 2014. <https://doi.org/10.48550/arXiv.1409.0685>
12. F. Zhu, Y. Wang, B. Fan, S. Xiang, G. Meng, C. Pan, Spectral unmixing via data-guided sparsity, *IEEE Trans. Image Process.*, **23** (2014), 5412–5427. <https://doi.org/10.1109/TIP.2014.2363423>
13. F. Zhu, Y. Wang, S. Xiang, B. Fan, C. Pan, Structured sparse method for hyperspectral unmixing, *ISPRS J. Photogramm. Remote Sens.*, **88** (2014), 101–118. <https://doi.org/10.1016/j.isprsjprs.2013.11.014>
14. J. Li, X. Li, L. Zhao, Hyperspectral unmixing via projected mini-batch gradient descent, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2017, 1133–1136. <https://doi.org/10.1109/IGARSS.2017.8127157>
15. F. Zhu, P. Honeine, Online nonnegative matrix factorization based on kernel machines, *2015 23rd European Signal Processing Conference (EUSIPCO)*, 2015, 2381–2385. <https://doi.org/10.1109/EUSIPCO.2015.7362811>
16. X. Li, F. Xu, Y. H. Dai, A stochastic proximal gradient method for linear hyperspectral unmixing, *J. Appl. Numer. Optim.*, **6** (2024), 323–338. <https://doi.org/10.23952/jano.6.2024.3.02>
17. Y. Qian, S. Jia, J. Zhou, A. Robles-Kelly, Hyperspectral unmixing via $l_{1/2}$ sparsity-constrained nonnegative matrix factorization, *IEEE Trans. Geosci. Remote Sens.*, **49** (2011), 4282–4297. <https://doi.org/10.1109/TGRS.2011.2144605>
18. Y. Dai, F. Xu, L. Zhang, Alternating direction method of multipliers for linear hyperspectral unmixing, *Math. Methods Oper. Res.*, **97** (2023), 289–310. <https://doi.org/10.1007/s00186-023-00815-2>



AIMS Press

©2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)