A SMOOTHING ACTIVE SET METHOD FOR LINEARLY CONSTRAINED NON-LIPSCHITZ NONCONVEX OPTIMIZATION [*]

CHAO ZHANG[†] AND XIAOJUN CHEN[‡]

Abstract. We propose a novel smoothing active set method for linearly constrained non-Lipschitz nonconvex problems. At each step of the proposed method, we approximate the objective function by a smooth function with a fixed smoothing parameter and employ a new active set method for minimizing the smooth function over the original feasible set, until a special updating rule for the smoothing parameter meets. The updating rule is always satisfied within finite number of iterations since the new active set method for smooth problems proposed in this paper forces at least one subsequence of projected gradients to zero. Any accumulation point of the smoothing active set method is a stationary point associated with the smoothing function used in the method, which is necessary for local optimality of the original problem. And any accumulation point for the $\ell_2 - \ell_p$ $(0 < p < 1)$ sparse optimization model is a limiting stationary point, which is a local minimizer under a certain second-order condition. Numerical experiments demonstrate the efficiency and effectiveness of our smoothing active set method for hyperspectral unmixing on 3D image cube of large size.

Key words. Non-Lipschitz, nonconvex, linearly constrained, smoothing active set method, stationary point

AMS subject classifications. 65K10 90C26 90C46

1. Introduction.

Active set methods have been successfully used for linearly constrained smooth optimization problems of large size; see [8, 13, 17, 18, 25, 42] and references therein. Hager and Zhang developed a novel active set algorithm for the bound constrained smooth optimization problems in [17], and ten years later they extended the method to solve linearly constrained smooth optimization problems [18]. The active set method in [18] switches between phase one that employs the gradient projection algorithm for the original problem and phase two that uses an algorithm with certain requirements for solving linearly constrained optimization problems on a face of the original feasible set. Hager and Zhang [18] showed that any accumulation point of the sequence generated by their method is a stationary point, and only phase two is performed after a finite number of iterations under certain conditions.

For linearly constrained nonsmooth convex optimization problems, Panier proposed an active set method [29], in which the search direction is computed by a bundle principle. And the convergence result is obtained under a certain nondegeneracy assumption. Wen et al. developed an active set algorithm for the unconstrained ℓ_1 minimization with good numerical performance and convergence results [36, 37]. For bound-constrained nonsmooth nonconvex optimization, Keskar and Wächter proposed a limited-memory quasi-Newton algorithm which uses an active set selection strategy to define the subspace in which search directions are computed [21]. Numerical experiments were conducted to show the efficacy of the algorithm, but theoretical convergence guarantees are elusive even for the unconstrained case. To the best of our knowledge, there is no active set method that tackles linearly constrained non-Lipschitz nonconvex optimization problems with solid convergence results.

---

[*]Submitted to the editors April 1, 2019.

Funding: C. Zhang's work was supported in part by NSFC grants 11571033, 11431002. X. Chen's work was supported in part by Hong Kong Research Council Grant PolyU15300/17P.

[†]Department of Applied Mathematics, Beijing Jiaotong University, Beijing 100044, China (zc.njtu@163.com).

[‡]Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hum, Kowloon, Hong Kong (xiaojun.chen@polyu.edu.hk).

One effective way to overcome the nonsmoothness in optimization is the type of smoothing methods, which uses the structure of the problem to define smoothing functions and the algorithms for solving smooth problems. Nesterov proposed a smoothing scheme [27] for minimizing a nonsmooth convex function over a convex set. Zhang and Chen proposed a smoothing projected gradient method [41] for minimizing a Lipschitz continuous function over a convex set. Bian and Chen developed a smoothing quadratic regularization method [4] for a class of linearly constrained non-Lipschitz optimization problems arising from image restoration. Xu et al. proposed a smoothing sequential quadratic programming method [38] for solving degenerate nonsmooth and nonconvex constrained optimization problems with applications to bilevel programs. Liu et al. proposed a smoothing sequential quadratic programming framework [26] for a class of composite ℓ_p $(0 < p < 1)$ minimization over polyhedron.

Inspired by the active set method [18] and the smoothing technique, we develop a novel smoothing active set method with solid convergence results for the following minimization problem

$$\min \; f(x) \quad \text{s.t.} \quad x \in \Omega, \tag{1.1}$$

where $f : R^n \to R$ is continuous but not necessarily Lipschitz continuous and

$$\Omega = \{x \in R^n \;:\; c_i^T x = d_i,\; i \in \mathcal{M}_E;\; c_i^T x \le d_i,\; i \in \mathcal{M}_I\}. \tag{1.2}$$

Here $\mathcal{M}_E = \{1, 2, \dots, m_e\}$, $\mathcal{M}_I = \{m_e + 1, m_e + 2, \dots, m\}$, $\mathcal{M} = \mathcal{M}_E \bigcup \mathcal{M}_I$, and $c_i \in R^n$, $d_i \in R$ for $i = 1, 2, \dots, m$.

Problem (1.1) involving a sparsity penalized term in the objective function has recently intrigued a lot of interests. It serves as a basic model for a variety of important applications, including the compressed sensing [1], the edge-preserving image restoration [4, 28], the sparse nonnegative matrix factorization for data classification [40], and the sparse portfolio selection [9, 15]. For example, the widely used $\ell_2 - \ell_p$ $(0 < p < 1)$ sparse optimization model

$$\min \|Ax - b\|^2 + \tau\|x\|_p^p \quad \text{s.t.} \quad x \ge 0, \tag{1.3}$$

where $\|\cdot\|$ refers to the Euclidean norm, $\|x\|_p^p = \sum_{i=1}^n |x_i|^p$, and $A \in R^{l\times n}$, $b \in R^l$, and $\tau > 0$ are given. The non-Lipschitz nonconvex term $\|x\|_p^p$ in the objective function and the nonnegative constraints benefit to recover some prior knowledge such as the sparsity of the signal, or the range of pixels. It is worth mentioning that in typical compressive sensing or image restoration, the dimension of optimization problems is large.

In order to develop the smoothing active set method, we first assume f is smooth in (1.1) in section 2 and develop an efficient new active set method for the linearly constrained smooth problems, which can be considered as a modification of the active set algorithm [18]. The new active set method combines the projected gradient (PG) method [8] and a linearly constrained optimizer (LCO) that satisfies mild requirements. We show in Theorem 2.2 that the new active set method forces at least one subsequence of projected gradients to zero. This property is essential in developing the smoothing active set method with global convergence in section 3. It is guaranteed that any accumulation point of the sequence generated by the new active set method is a stationary point. Moreover, if the sequence generated by the new active set method converges to a stationary point x^*, then the sequence can identify the set of strongly active constraints and hence is trapped by the face exposed by $-\nabla f(x^*)$

after a finite number of iterations. The convergence and identification properties are not guaranteed by the active set method in [18] for the smooth problems. Based on the identification properties, we also prove the local convergence result that if the sequence converges to x^* and the strong second-order sufficient optimality condition holds, then only the LCO is executed after a finite number of iterations.

Combining the new active set method for linearly constrained smooth minimization problem with delicate smoothing strategies, we then develop in section 3 a novel smoothing active set method that solves the linearly constrained non-Lipschitz minimization problem (1.1). The new active set method for smooth problems is used to solve the smoothing problems. We give the concept of a stationary point associated with the smoothing function and show that it is necessary for optimality of the original problem. We show that any accumulation point generated by the smoothing active set method is a stationary point of the original problem. Moreover, it is a limiting stationary point of problem (1.3). If in addition a second-order condition holds, it is also a strict local minimizer of (1.3).

We conduct numerical experiments on real applications of large scale in hyperspectral unmixing in section 4. The numerical results manifest that the smoothing active set method performs favorably in comparison to several state-of-the-art methods in hyperspectral unmixing.

Throughout the paper, we use the following notation. $\langle x, y\rangle = x^T y$ presents the inner product of two vectors x and y of the same dimension. $R^n_+ = \{x \in R^n \ : \ x \geq 0\}$ and $R^n_{++} = \{x \in R^n \ : \ x > 0\}$. $|\mathcal{S}|$ corresponds to the cardinality of a finite set $\mathcal{S}$. If $\mathcal{S}$ is a subset of $\{1, 2, \ldots, n\}$, then for any vector $u \in R^n$ and $M \in R^{n\times n}$, $u_{\mathcal{S}}$ is the subvector of u whose entries lie in u indexed by $\mathcal{S}$, and $M_{\mathcal{SS}}$ denotes the submatrix of M whose rows and columns lie in $\mathcal{S}$. $\mathcal{N}(M)$ is the null space of M. Let $\mathbb{N}$ be the set of all natural numbers and $\mathcal{N}^\sharp_\infty$ be the infinite subsets of $\mathbb{N}$. We use the notation $\underset{N}{\rightarrow}$ for the convergence indexed by $N \in \mathcal{N}^\sharp_\infty$. The normal cone to a closed convex set Ω at x is denoted by $N_\Omega(x)$, and $P_\Omega[x] = \text{argmin}\{\|z - x\| \ : \ z \in \Omega\}$ is the orthogonal projection from x into Ω. The ball with center x^* and radius δ is denoted by $B_\delta(x^*)$. For any $x \in R^n$, the active and free index sets are defined by

$$\mathcal{A}(x) := \mathcal{M}_E \cup \{i \in \mathcal{M}_I \ : \ c_i^T x = d_i\}, \quad \mathcal{F}(x) := \{i \in \mathcal{M}_I \ : \ c_i^T x < d_i\}.$$

2. A new active set method for linearly constrained smooth minimization. In this section, we consider the following linearly constrained smooth problem

$$\min \ f(x) \quad \text{s.t.} \quad x \in \Omega, \tag{2.1}$$

where f is continuously differentiable and Ω is defined in (1.2).

Recall that the projected gradient $\nabla_\Omega f(x)$ is defined by

$$\nabla_\Omega f(x) \equiv P_{T(x)}[-\nabla f(x)] = \text{argmin}\{\|v + \nabla f(x)\| \ : \ v \in T(x)\},$$

where $T(x)$ is the tangent cone to Ω at x. Calamai and Moré (Lemma 3.1 of [8]) showed that $x^* \in \Omega$ is a stationary point of (2.1) if and only if $\nabla_\Omega f(x^*) = 0$. It is worth mentioning that $\|\nabla_\Omega f(x)\|$ can be bounded away from zero in a neighborhood of a stationary point x^*, since $\|\nabla_\Omega f(\cdot)\|$ is not continuous, but only lower semicontinuous on Ω according to Lemma 3.3 of [8]. That is, for any $\{x^k\} \subset \Omega$ converging to x,

$$\|\nabla_\Omega f(x)\| \leq \liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\|.$$

A stationary point x^* of (2.1) is often characterized as

$$d^1(x^*) := P_\Omega[x^* - \nabla f(x^*)] - x^* = 0.$$

We find that convergence of most existing active set methods for (2.1) is to show $\liminf_{k\to\infty} \|d^1(x^k)\| = 0$, such as the active set method in [18]. However, since the norm of projected gradient is not continuous, $\liminf_{k\to\infty} \|d^1(x^k)\| = 0$ does not imply $\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$. See Example 1 in section 2. The new active set method proposed in this section aims to have

$$\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0,$$

which is essential for showing the convergence result of the smoothing active set method for solving nonsmooth problem (1.1) proposed in section 3.

2.1. Structure of the new active set method. Now we introduce the necessary notation used in the new active set method. Let us denote $g(x) = \nabla f(x)$. Given an index set $\mathcal{S}$ satisfying $\mathcal{M}_E \subseteq \mathcal{S} \subseteq \mathcal{M}$, we define $g^{\mathcal{S}}(x) \in R^n$ by

$$g^{\mathcal{S}}(x) = P_{\mathcal{N}(C_{\mathcal{S}}^T)}[g(x)] = \arg\min\{\|y - g(x)\| : y \in R^n \text{ and } C_{\mathcal{S}}^T y = 0\}, \tag{2.2}$$

where $C_{\mathcal{S}} \in R^{n\times|\mathcal{S}|}$ is the matrix whose columns are $c_i, i \in \mathcal{S}$. In particular, we denote $g^{\mathcal{A}}(x)$ for $\mathcal{S} = \mathcal{A}(x)$ and if $\mathcal{A}(x) = \emptyset$, then $g^{\mathcal{A}}(x) = g(x)$. Thus $g^{\mathcal{A}}(x)$ is the unique optimal solution of the strongly convex problem

$$\min \quad \frac{1}{2}\|y - g(x)\|^2 \quad \text{s.t.} \quad c_i^T y = 0,\ i \in \mathcal{A}(x). \tag{2.3}$$

From the first-order optimality conditions, it is easy to find that for $x \in \Omega$, $g^{\mathcal{A}}(x) = 0$ if and only if x is a stationary point of f on its associated face

$$\breve{\Omega}(x) := \{y \in \Omega \ : \ c_i^T y = d_i \text{ for all } i \in \mathcal{A}(x)\}. \tag{2.4}$$

Let x^* be a stationary point of (2.1) and $\Lambda(x^*)$ be the set of Lagrange multipliers associated with the constraints. That is, $x^* \in \Omega$ and for any $\lambda^* \in \Lambda(x^*)$, (x^*, λ^*) satisfies

$$\begin{gathered} g(x^*) + \textstyle\sum_{i\in\mathcal{M}} \lambda_i^* c_i = 0, \\ \lambda_i^* \geq 0 \text{ if } i \in \mathcal{M}_I \cap \mathcal{A}(x^*), \qquad \lambda_i^* = 0 \text{ if } i \in \mathcal{F}(x^*), \\ \lambda_i^*(c_i^T x^* - d_i) = 0 \text{ for all } i \in \mathcal{M}_I. \end{gathered} \tag{2.5}$$

Consider

$$y(x,\alpha) = P_\Omega[x - \alpha g(x)] = \operatorname{argmin}\left\{\|x - \alpha g(x) - y\|^2 \ : \ y \in \Omega\right\}, \tag{2.6}$$

where $\alpha > 0$ is a given number. Thus there exists $\lambda \in R^m$ such that $(y(x,\alpha), \lambda)$ satisfies

$$\begin{gathered} y(x,\alpha) - (x - \alpha g(x)) + \textstyle\sum_{i\in\mathcal{M}} \lambda_i c_i = 0, \\ \lambda_i \geq 0 \text{ if } i \in \mathcal{M}_I \cap \mathcal{A}(y(x,\alpha)), \qquad \lambda_i = 0 \text{ if } i \in \mathcal{F}(y(x,\alpha)), \\ \lambda_i(c_i^T y(x,\alpha) - d_i) = 0 \text{ for all } i \in \mathcal{M}_I. \end{gathered} \tag{2.7}$$

Let $\Lambda(x,\alpha)$ be the set of Lagrange multipliers satisfying (2.7) at the solution $y = y(x,\alpha)$ of (2.6). It is easy to see that

$$y(x^*,\alpha) = x^* \quad \text{and} \quad \Lambda(x^*,\alpha) = \alpha\Lambda(x^*). \tag{2.8}$$

In the new active set method, it employs either the iteration of the PG method or the iteration of the LCO by given rules. Let x^k be the current iterate and the LCO be chosen to get the new iterate. Then the LCO solves the problem

$$\min \quad f(y) \quad \text{s.t.} \quad y \in \breve{\Omega}(x^k), \tag{2.9}$$

which operates on the faces of Ω. Compared to the original problem (2.1), there are usually much more equality constraints in (2.9) which may lead the efficiency of the LCO. This is obviously true when the feasible set is defined by the bound constraints or the simplex constraint (which are sometimes called "hard constraints" and it is better to satisfy them strictly rather than penalize them into the objective function). The PG step comes from the classic "piecewise PG method" proposed in [8], and an arbitrary LCO can be chosen as long as it satisfies certain requirements listed below.

- PG method
  Given $\rho, \beta \in (0,1)$. For $k = 1, 2, \dots,$
  set $d^k = -g(x^k)$ and let $x^{k+1} = P_\Omega[x^k + \alpha_k d^k]$ where α_k is determined by the Armijo line search, i.e., $\alpha_k = \max\{\rho^0, \rho^1, \dots\}$ is chosen such that

$$f(x^{k+1}) \le f(x^k) + \beta\langle g(x^k), x^{k+1} - x^k\rangle. \tag{2.10}$$

- LCO Requirements
  For $k = 1, 2, \dots,$
  F1: $x^k \in \Omega$ and $f(x^{k+1}) \le f(x^k)$ for each k.
  F2: $\mathcal{A}(x^k) \subseteq \mathcal{A}(x^{k+1})$ for each k.
  F3: If $\exists\ \bar{k} > 0$ such that $\mathcal{A}(x^j) \equiv \bar{\mathcal{A}}$ for all $j \ge \bar{k}$, then $\liminf_{j\to\infty} \|g^{\mathcal{A}}(x^j)\| = 0$.

F1 and F2 of the LCO Requirements are satisfied, as long as the LCO adopts a monotone line search, and whenever a new constraint becomes active, it changes the corresponding inequality constraint to the equality constraint in (2.9). Later we always assume the two strategies are incorporated in the LCO. F3 requires that if the active set becomes stable as $\mathcal{A}(x^j) \equiv \bar{\mathcal{A}}$, then at least one accumulation point x^* of the sequence $\{x^k\}$ generated by the LCO is a stationary point of problem (2.9) with $\breve{\Omega}(x^k) = \{y \in \Omega \ : \ c_i^T y = d_i \text{ for all } i \in \bar{\mathcal{A}}\}$. Note that in this case x^* is a stationary point if and only if $g^{\bar{\mathcal{A}}}(x^*) = 0$. And since $g^{\bar{\mathcal{A}}}(x) = P_{\mathcal{N}(C_{\bar{\mathcal{A}}}^T)}[g(x)]$, we know that $g^{\bar{\mathcal{A}}}(\cdot) : R^n \to R^n$ is a continuous function. Thus $g^{\bar{\mathcal{A}}}(x^*) = 0$ indicates

$$\liminf_{j\to\infty} \|g^{\mathcal{A}}(x^j)\| = \liminf_{j\to\infty} \|g^{\bar{\mathcal{A}}}(x^j)\| = 0.$$

Therefore the LCO Requirements can be easily fulfilled by many algorithms based on gradient or Newton type iterations that employ a monotone line search and add constraints to the active set whenever a new constraint becomes active, e.g., the projected gradient method [8], the method of Zoutendijk (section 10.1 of [2]), the Frank-Wolfe algorithm [16], the first-order interior-point method [33], and the affine-scaling interior-point method [19]. When $\Omega = R^n_+$, we can employ the LCO using essentially unconstrained optimization methods such as the conjugate gradient method as in [17].

Now we are ready to outline the new active set method for problem (2.1).

2.2. Convergence analysis.

Assumption 2.1. For any $\Gamma \in R$, the level set

$$\mathcal{L}_\Gamma = \{x \in \Omega \ : \ f(x) \le \Gamma\}$$

is bounded.

Algorithm 2.1 A new active set method

```
Parameters: ϵ ∈ [0, ∞), θ and η ∈ (0, 1). x^1 = P_Ω[x^0], k = 1.
Phase one:
while ‖∇_Ω f(x^k)‖ > ϵ, do
    Execute the PG step to obtain x^{k+1} from x^k. Let k ← k + 1.
    If ‖g^A(x^k)‖ ≤ θ‖∇_Ω f(x^k)‖, then θ ← ηθ.
    If ‖g^A(x^k)‖ > θ‖∇_Ω f(x^k)‖, then go to phase two.
end while
Phase two:
while ‖∇_Ω f(x^k)‖ > ϵ, do
    Execute the LCO step to obtain x^{k+1} from x^k. Let k ← k + 1.
    If ‖g^A(x^k)‖ ≤ θ‖∇_Ω f(x^k)‖, then go to phase one and θ ← ηθ.
end while
```

In the remainder of this paper, we assume that the LCO satisfies the LCO Requirements F1-F3, and Assumption 2.1 holds. We now show the global convergence of Algorithm 2.1 for problem (2.1).

THEOREM 2.2. *Let $\{x^k\}$ be the sequence generated by Algorithm 2.1 with $\epsilon = 0$. Then there exists at least one accumulation point of $\{x^k\}$,*

$$\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0, \tag{2.11}$$

and any accumulation point of $\{x^k\}$ is a stationary point of (2.1).

Proof. By Assumption 2.1, there exists at least one accumulation point x^* of $\{x^k\}$. Let $\{x^k\}_{k\in K}$ be an infinite subsequence of $\{x^k\}$ such that $\lim\limits_{k\to\infty,\ k\in K} x^k = x^*$.

If only phase one is performed for k sufficiently large, then by Assumption 2.1 and Theorem 2.4 of [8],

$$\lim_{k\to\infty,\ k\in K} \frac{x^{k+1} - x^k}{\alpha_k} = 0.$$

Hence for $k \to \infty$, $k \in K$,

$$\|x^{k+1} - x^*\| \le \|x^{k+1} - x^k\| + \|x^k - x^*\| \to 0,$$

which indicates $\lim_{k\to\infty,\ k\in K} x^{k+1} = x^*$. According to Theorem 3.4 of [8],

$$\lim_{k\to\infty,\ k\in K} \|\nabla_\Omega f(x^{k+1})\| = 0.$$

By the lower semicontinuity of $\|\nabla_\Omega f(\cdot)\|$ shown in Lemma 3.3 of [8],

$$\|\nabla_\Omega f(x^*)\| \le \lim_{k\to\infty,\ k\in K} \|\nabla_\Omega f(x^{k+1})\| = 0,$$

which guarantees that x^* is a stationary point of (2.1).

If only phase two is performed for k sufficiently large, then there exists $\hat\theta > 0$ such that $\theta \equiv \hat\theta$ for k sufficiently large, because θ is never reduced in phase two. Hence for k sufficiently large,

$$\|g^{\mathcal{A}}(x^k)\| \ge \hat\theta \|\nabla_\Omega f(x^k)\|. \tag{2.12}$$

Note that the LCO works on the faces of Ω and no index in the active set can be freed from x^k to x^{k+1} using the LCO. By F2 of the LCO Requirements, the active set becomes stable for k large enough and hence $\liminf_{k\to\infty} \|g^{\mathcal{A}}(x^k)\| = 0$ according to F3. From (2.12) we then have (2.11) holds. By the lower semicontinuity of $\|\nabla_\Omega f(\cdot)\|$, x^* is a stationary point of (2.1).

The remaining case is that there are an infinite number of branches from phase two to phase one for $\{x^k\}_{k\in K}$. Then phase one is performed an infinite number of times at $k_1 < k_2 < \cdots < \cdots$, where $\{k_i\} \subseteq K$. By Theorem 3.4 of [8], $\lim_{k_i\to\infty} \|\nabla_\Omega f(x^{k_i+1})\| = 0$. Again we find x^* is a stationary point by using $\{x^{k_i+1}\} \to x^*$ and the lower semicontinuity of $\|\nabla_\Omega f(\cdot)\|$. The proof is completed. □

Identification properties of an algorithm for linearly constrained problems are significant from both a theoretical and a practical point of view [14, 25]. For a stationary point x^*, the set of strongly active constraints is defined by

$$\mathcal{A}_+(x^*) = \mathcal{M}_E \cup \{i \in \mathcal{M}_I \ : \ c_i^T x^* = d_i, \text{ and } \exists \lambda^* \in \Lambda(x^*) \text{ such that } \lambda_i^* > 0\}.$$

In convex analysis, the face of a convex set Ω exposed by the vector $w \in R^n$ is

$$E[w] \equiv \operatorname{argmax}\{w^T x \ : \ x \in \Omega\}.$$

A computation based on the definition of a face shows that for the polyhedral set Ω given in (1.2),

$$E[-\nabla f(x^*)] = \{x \in \Omega \ : \ c_i^T x = d_i \ \text{ if } \lambda_i^* > 0 \text{ for } i \in \mathcal{M}_I\}, \tag{2.13}$$

where $\lambda^* \in \Lambda(x^*)$. Note that this expression is valid for any choice of Lagrange multipliers $\lambda^* \in \Lambda(x^*)$.

We say that the linear independence constraint qualification (LICQ) holds at a point $x \in \Omega$, if the gradients $c_i, i \in \mathcal{A}(x)$ are linearly independent.

THEOREM 2.3. *Let $\{x^k\}$ be a sequence generated by Algorithm 2.1 with $\epsilon = 0$ which converges to x^*. Suppose that the LICQ holds at x^*, and for some $\delta > 0$, g is Lipschitz continuous in $B_\delta(x^*)$ with a Lipschitz constant ϱ. Then there exists an integer $\hat{k}_0 > 0$ such that*

$$\mathcal{A}_+(x^*) \subseteq \mathcal{A}(x^k) \quad \textit{and} \quad x^k \in E[-\nabla f(x^*)], \qquad \textit{for } k \geq \hat{k}_0.$$

Proof. Since $\{x^k\}$ converges to x^*, there exists $k_0 > 0$ such that $x^k \in B_\delta(x^*)$ for any $k \geq k_0$. Using the definition of $y(x,\alpha)$ in (2.6) and the Lipschitz continuity of g with the Lipschitz constant ϱ in $B_\delta(x^*)$, we have for any $\alpha > 0$ and $k \geq k_0$,

$$\begin{aligned}
\|y(x^k,\alpha) - x^*\| &= \|y(x^k,\alpha) - y(x^*,\alpha)\| \\
&= \|P_\Omega[x^k - \alpha g(x^k)] - P_\Omega[x^* - \alpha g(x^*)]\| \\
&\leq \|x^k - x^* + \alpha(g(x^*) - g(x^k))\| \\
&\leq (1 + \alpha\varrho)\|x^k - x^*\|.
\end{aligned}$$

Since $\{x^k\}$ converges to x^*, there is an integer $\bar{k} > 0$ such that $\mathcal{F}(x^*) \subseteq \mathcal{F}(y(x^k,\alpha))$ for $k \geq \bar{k}$. We know that $\Lambda(x^*)$ is a singleton, since the gradients of the active constraints at x^* are linearly independent. Thus $\Lambda(x^*,\alpha) = \alpha\Lambda(x^*)$ is also a singleton for any given $\alpha > 0$. Moreover, $\Lambda(x^k,\alpha)$ is a singleton for $k \geq \bar{k}$, because $\mathcal{A}(y(x^k,\alpha)) \subseteq \mathcal{A}(x^*)$ for $k \geq \bar{k}$ and the gradients of the active constraints at $y(x^k,\alpha)$ are linearly independent.

Consider the linear system

$$q + \sum_{i\in\mathcal{M}} \lambda_i c_i = 0, \quad \lambda_i \ge 0 \text{ for } i \in \mathcal{M}_I,\ \lambda_i = 0 \text{ for } i \in \mathcal{F}(x^*). \tag{2.14}$$

Let

$$p_1 = y(x^k,\alpha) - x^k + \alpha g(x^k), \quad \text{and} \quad p_2 = y(x^*,\alpha) - x^* + \alpha g(x^*).$$

According to (2.7), $\lambda^k \in \Lambda(x^k,\alpha)$ is feasible in the linear system (2.14) with $q = p_1$. And by (2.5) and (2.8), it is easy to see that for $\lambda^* \in \Lambda(x^*)$, $\alpha\lambda^* \in \Lambda(x^*,\alpha)$ is also feasible in the same system (2.14) but with $q = p_2$. Hence by Hoffman's result (see, e.g., Theorem 7.12 of [32]) and the fact that $\Lambda(x^*,\alpha)$ is a singleton, there exists a positive constant ν, independent of p_1 and p_2 and depending only on c_i, $i \in \mathcal{M}$, such that

$$\|\lambda^k - \alpha\lambda^*\| \le \nu\|p_1 - p_2\| \le 2\nu(1+\alpha\varrho)\|x^k - x^*\|.$$

For any $i_0 \in \mathcal{M}_I \cap \mathcal{A}_+(x^*)$, the Lagrange multiplier $\lambda^* \in \Lambda(x^*)$ satisfies $\lambda^*_{i_0} > 0$. Thus there exists an integer $\tilde{k}_{i_0} > 0$ such that $\lambda^k_{i_0} > 0$ for all $k \ge \tilde{k}_{i_0}$. Now we consider (2.6) and its first-order optimality conditions given in (2.7). We find that $c_{i_0}^T y(x^k,\alpha) = d_{i_0}$ by complementarity and hence $i_0 \in \mathcal{A}(y(x^k,\alpha))$. Let

$$\tilde{k} = \max\{\tilde{k}_i,\ i \in \mathcal{M}_I \cap \mathcal{A}_+(x^*)\} \quad \text{and} \quad \hat{k} = \max\{\bar{k},\tilde{k}\}.$$

Clearly for any $i \in \mathcal{A}_+(x^*)$ and any given $\alpha > 0$,

$$i \in \mathcal{A}(y(x^k,\alpha)) \quad \text{for all } k \ge \hat{k}.$$

We need to consider two possible cases.

Case 1: There exists an integer $\hat{k}_1 \ge \hat{k}$ such that $x^{\hat{k}_1+1}$ is obtained from the PG step in Algorithm 2.1. Then for any $k \ge \hat{k}_1$ such that x^{k+1} is obtained from x^k by the PG step in Algorithm 2.1, we know by (2.6)

$$x^{k+1} = P_\Omega[x^k - \alpha_k g(x^k)] = y(x^k,\alpha_k),$$

and consequently $i \in \mathcal{A}(x^{k+1})$ for any $i \in \mathcal{A}_+(x^*)$. Since no active constraint can be freed by the LCO step in phase two, we get

$$i \in \mathcal{A}(x^k) \quad \text{for any } k \ge \hat{k}_1 + 1.$$

Case 2: x^{k+1} is obtained from the LCO step in phase two for any $k \ge \hat{k}$. By F2 of the LCO Requirements, we find $\mathcal{A}(x^k) \subseteq \mathcal{A}(x^{k+1})$ for all $k \ge \hat{k}$. Then the active constraints become unchanged after a finite number of steps. Thus there exists an integer $\hat{k}_2 > \hat{k}$ such that

$$\mathcal{A}(x^k) \equiv \tilde{\mathcal{A}} \subseteq \mathcal{A}(x^*) \text{ for all } k \ge \hat{k}_2.$$

By the definition of $g^{\tilde{\mathcal{A}}}(x^k)$, and the first-order optimality conditions at the global optimizer $g^{\tilde{\mathcal{A}}}(x^k)$, there exists a unique vector $\pi^k \in R^m$ such that

$$\begin{array}{c} g^{\tilde{\mathcal{A}}}(x^k) - g(x^k) - \sum_{i\in\tilde{\mathcal{A}}} \pi^k_i c_i = 0, \\ c_i^T g^{\tilde{\mathcal{A}}}(x^k) = 0,\ i \in \tilde{\mathcal{A}}, \quad \pi^k_i = 0 \text{ if } i \notin \tilde{\mathcal{A}}. \end{array} \tag{2.15}$$

Here the vector π^k is unique because the column vectors c_i, $i \in \tilde{\mathcal{A}} \subseteq \mathcal{A}(x^*)$ are linearly independent. Similarly, by the strong convexity of problem (2.3) with x

being replaced by x^*, and the linear independence of $\{c_i, i \in \mathcal{A}(x^*)\}$, there exist a unique vector $g^{\mathcal{A}}(x^*) \in R^n$ and a unique vector $\lambda \in R^m$ such that

$$\begin{aligned} g^{\mathcal{A}}(x^*) - g(x^*) - \textstyle\sum_{i \in \mathcal{A}(x^*)} \lambda_i c_i = 0, \\ c_i^T g^{\mathcal{A}}(x^*) = 0,\ i \in \mathcal{A}(x^*), \quad \lambda_i = 0 \text{ if } i \notin \mathcal{A}(x^*). \end{aligned} \tag{2.16}$$

And there exists a unique vector $\lambda^* \in R^m$ such that

$$g(x^*) = -\sum_{i \in \mathcal{A}(x^*)} \lambda_i^* c_i, \quad \lambda_i^* = 0 \text{ if } i \notin \mathcal{A}(x^*), \tag{2.17}$$

since x^* is a stationary point of (2.1) and the gradients of the active constraints at x^* are linearly independent.

We get $g^{\mathcal{A}}(x^*) = 0$ and $\lambda = \lambda^*$, by comparing (2.16), (2.17) and using the uniqueness of $g^{\mathcal{A}}(x^*)$ and λ in (2.16). Moreover, $\liminf_{k \to \infty} g^{\tilde{A}}(x^k) = 0$ according to F3 of the LCO Requirements. Let $\{k_j\} \subseteq \{k\}$ be an infinite subsequence such that $\lim_{k_j \to \infty} g^{\tilde{A}}(x^{k_j}) = 0$. Taking limit to the first linear system in (2.15), we have

$$0 = \lim_{k_j \to \infty} g^{\tilde{A}}(x^{k_j}) = g(x^*) + \sum_{i \in \tilde{A}} \lim_{k_j \to \infty} \pi_i^{k_j} c_i. \tag{2.18}$$

Comparing (2.17) and (2.18), and noting the uniqueness of λ^* in (2.17), we find

$$\lim_{k_j \to \infty} \pi_i^{k_j} = \lambda_i^* > 0 \quad \text{for any } i \in \mathcal{A}_+(x^*) \setminus \mathcal{M}_E.$$

Since $\pi_i^k = 0$ if $i \notin \tilde{A}$ for k sufficiently large according to (2.15), we know

$$\lim_{k_j \to \infty} \pi_i^{k_j} = 0 \quad \text{for any } i \in \mathcal{A}_+(x^*) \setminus \tilde{A}.$$

This indicates $\mathcal{A}_+(x^*) \setminus \tilde{A} = \emptyset$. Hence for any $i \in \mathcal{A}_+(x^*)$, we get $i \in \tilde{A} \equiv \mathcal{A}(x^k)$ for $k \geq \hat{k}_2$.

Thus in any case, there exists an index $\hat{k}_0$ ($\hat{k}_0 = \hat{k}_1 + 1$ if Case 1 occurs, and $\hat{k}_0 = \hat{k}_2$ if Case 2 happens otherwise) such that

$$\mathcal{A}_+(x^*) \subseteq \mathcal{A}(x^k) \quad \text{for } k \geq \hat{k}_0.$$

This, combined with (2.13), implies

$$x^k \in E[-\nabla f(x^*)] \quad \text{for } k \geq \hat{k}_0.$$

We complete the proof. ∎

Based on the identification properties analyzed above, we will show the local convergence result that only iterations in phase two occur for k sufficiently large, if we further assume that the strong second-order sufficient optimality condition holds at x^*. A stationary point x^* of (2.1) satisfies the strong second-order sufficient optimality condition if there exists $\sigma > 0$ such that

$$v^T \nabla^2 f(x^*) v \geq \sigma \|v\|^2 \tag{2.19}$$

for all $v \in R^n$ such that $c_i^T v = 0$ for all $i \in \mathcal{A}_+(x^*)$.

LEMMA 2.4. *Let $\{x^k\}$ be a sequence generated by Algorithm 2.1 with $\epsilon = 0$ which converges to x^*. Suppose that the LICQ holds at x^*, and for some $\delta > 0$, g is Lipschitz continuous in $B_\delta(x^*)$ with a Lipschitz constant ϱ. Then*

$$\|\nabla_\Omega f(x^k)\| \le \varrho \|x^k - x^*\| \quad \text{for } k \text{ sufficiently large.} \tag{2.20}$$

Proof. From the nonexpansive property of the projection operator,

$$\begin{aligned} \|\nabla_\Omega f(x^k)\| &= \|P_{T(x^k)}[-g(x^k)] - P_{T(x^k)}[-g(x^*)] + P_{T(x^k)}[-g(x^*)]\| \\ &\le \|g(x^k) - g(x^*)\| + \|P_{T(x^k)}[-g(x^*)]\|. \end{aligned} \tag{2.21}$$

Similarly,

$$\begin{aligned} &\|P_{T(x^k)}[-g(x^*)]\| \\ &\quad = \|P_{T(x^k)}[-g(x^*)] - P_{T(x^k)}[-g(x^k)] + P_{T(x^k)}[-g(x^k)]\| \\ &\quad \le \|g(x^k) - g(x^*)\| + \|\nabla_\Omega f(x^k)\|. \end{aligned} \tag{2.22}$$

From (2.21) and (2.22),

$$\|\nabla_\Omega f(x^k)\| - \|g(x^k) - g(x^*)\| \le \|P_{T(x^k)}[-g(x^*)]\| \le \|\nabla_\Omega f(x^k)\| + \|g(x^k) - g(x^*)\|.$$

Theorem 2.3 guarantees that there is an integer $\hat{k}_0$ such that $x^k \in E[-\nabla f(x^*)]$ for all $k \ge \hat{k}_0$. Thus according to Theorem 3.1 of [25], $\lim_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$. This, combined with (2.22) and the facts that $\{x^k\} \to x^*$ and g is locally Lipschitz continuous at x^*, yields

$$\lim_{k\to\infty} \|P_{T(x^k)}[-g(x^*)]\| = 0. \tag{2.23}$$

By direct computation,

$$T(x^k) = \{v \ : \ c_i^T v = 0, \ i \in \mathcal{M}_E; \ \ c_i^T v \le 0, \ i \in \mathcal{M}_I \cap \mathcal{A}(x^k)\}. \tag{2.24}$$

When x^k is sufficiently near x^*, we know $\mathcal{F}(x^*) \subseteq \mathcal{F}(x^k)$. Then by Theorem 2.3, we find

$$\mathcal{A}_+(x^*) \subseteq \mathcal{A}(x^k) \subseteq \mathcal{A}(x^*). \tag{2.25}$$

From the inclusions in (2.25) and the fact that $\mathcal{A}(x^*)$ has finite number of subsets, there are only a finite number of index sets $\mathcal{A}_1, \ldots, \mathcal{A}_\nu$ for $\mathcal{A}(x^k)$, $k = 1, 2, \ldots$. From the expression of $T(x^k)$ in (2.24), let us define

$$T_j = \{v \ : \ c_i^T v = 0, \ i \in \mathcal{M}_E; \ \ c_i^T v \le 0, \ i \in \mathcal{M}_I \cap \mathcal{A}_j\} \quad \text{for} \quad j = 1, 2, \ldots, \nu.$$

Without loss of generality, we assume

$$\{T_1, T_2, \ldots, T_t\} \subseteq \{T_1, T_2, \ldots, T_\nu\}$$

is composed by all the elements in $\{T_1, T_2, \ldots, T_\nu\}$ such that each T_j, $j = 1, 2, \ldots, t$, contains an infinite number of $T(x^k)$, $k = 1, 2, \ldots$. Hence we get $P_{T_j}[-g(x^*)] = 0$ for $j = 1, 2, \ldots, t$, according to (2.23). Consequently, for all k sufficiently large, we have

$$P_{T(x^k)}[-g(x^*)] \in \{P_{T_1}[-g(x^*)], P_{T_2}[-g(x^*)], \ldots, P_{T_t}[-g(x^*)]\},$$

which indicates

$$P_{T(x^k)}[-g(x^*)] = 0 \quad \text{for all } k \text{ sufficiently large.} \tag{2.26}$$

Substituting (2.26) into (2.21) and using the Lipschitz continuity of g with the Lipschitz constant ϱ in $B_\delta(x^*)$, we get our desired result (2.20). ∎

LEMMA 2.5. *Let $\{x^k\}$ be a sequence generated by Algorithm 2.1 with $\epsilon = 0$ which converges to x^*. If f is twice continuously differentiable in a neighborhood of x^*, the LICQ holds at x^*, and the strong second-order sufficient optimality condition in (2.19) holds at x^*, then there exists $\theta^* > 0$ such that*

$$\|g^{\mathcal{A}}(x^k)\| \geq \theta^* \|\nabla_\Omega f(x^k)\| \quad \textit{for all } k \textit{ sufficiently large.} \tag{2.27}$$

Proof. By Theorem 2.3, $\mathcal{A}_+(x^*) \subseteq \mathcal{A}(x^k)$ for $k \geq k_0$. Thus $x^k - x^*$ satisfies $c_i^T(x^k - x^*) = 0$ for all $i \in \mathcal{A}_+(x^*)$ and $k \geq k_0$. By the strong second-order sufficient optimality condition, we find x^* is a strict local minimizer of (2.1), and for k sufficiently large,

$$(x^k - x^*)^T(g(x^k) - g(x^*)) \geq 0.5\sigma\|x^k - x^*\|^2. \tag{2.28}$$

Using the first-order necessary optimality conditions for a local minimizer of (2.1), we know that there exists a multiplier $\lambda^* \in R^m$ such that

$$g(x^*) + \sum_{i\in\mathcal{M}} \lambda_i^* c_i = 0, \quad (d_i - c_i^T x^*)\lambda_i^* = 0,\ i \in \mathcal{M}; \quad \lambda_i^* \geq 0,\ i \in \mathcal{M}_I. \tag{2.29}$$

We have for k sufficiently large, $\mathcal{A}_+(x^*) \subseteq \mathcal{A}(x^k)$ and $d_i - c_i^T x^* = 0 = d_i - c_i^T x^k$ when $i \in \mathcal{A}_+(x^*)$, and $\lambda_i^* = 0$ when $i \notin \mathcal{A}_+(x^*)$. Hence

$$\lambda_i^* c_i^T(x^k - x^*) = \lambda_i^*[(d_i - c_i^T x^*) - (d_i - c_i^T x^k)] = 0 \quad \text{for all } i \in \mathcal{M}.$$

This, combined with (2.29), yields

$$(x^k - x^*)^T g(x^*) = (x^k - x^*)^T[g(x^*) + \sum_{i\in\mathcal{M}} \lambda_i^* c_i] = 0. \tag{2.30}$$

Denote here $\mathcal{S} = \mathcal{A}(x^k)$ for simplicity. The first-order optimality conditions for the minimizer $g^{\mathcal{S}}(x^k)$ in (2.2) implies the existence of $\lambda_{\mathcal{S}} \in R^{|\mathcal{S}|}$ such that

$$g^{\mathcal{S}}(x^k) - g(x^k) + C_{\mathcal{S}}\lambda_{\mathcal{S}} = 0. \tag{2.31}$$

Because $\mathcal{A}(x^k) \subseteq \mathcal{A}(x^*)$ for $k \geq k_0$, we have $c_i^T(x^k - x^*) = 0$ for all $i \in \mathcal{S}$. Hence

$$[C_{\mathcal{S}}^T(x^k - x^*)]^T \lambda_S = 0, \quad \text{for all } k \text{ sufficiently large.} \tag{2.32}$$

By (2.31) and (2.32), we find

$$(x^k - x^*)^T g(x^k) = (x^k - x^*)^T[g^{\mathcal{S}}(x^k) + C_{\mathcal{S}}\lambda_{\mathcal{S}}] = (x^k - x^*)^T g^{\mathcal{S}}(x^k). \tag{2.33}$$

Using the Cauchy-Schwarz inequality, (2.33), (2.28) and (2.30) sequentially, we get

$$\begin{aligned}
\|x^k - x^*\|\|g^{\mathcal{S}}(x^k)\| &\geq (x^k - x^*)^T g^{\mathcal{S}}(x^k) \\
&= (x^k - x^*)^T g(x^k) \\
&= (x^k - x^*)^T[g(x^k) - g(x^*) + g(x^*)] \\
&\geq 0.5\sigma\|x^k - x^*\|^2.
\end{aligned}$$

Reminding that $\mathcal{S} = \mathcal{A}(x^k)$, we have

$$\|g^{\mathcal{A}}(x^k)\| \geq 0.5\sigma\|x^k - x^*\| \quad \text{for } k \text{ sufficiently large.} \tag{2.34}$$

This, together with Lemma 2.4, deduces (2.27) with $\theta^* = 0.5\dfrac{\sigma}{\varrho}$. ∎

We are ready to show that the new active set method given in Algorithm 2.1 will only perform the LCO within a finite number of iterations.

THEOREM 2.6. *Let $\{x^k\}$ be a sequence generated by Algorithm 2.1 with $\epsilon = 0$ which converges to x^*. If the assumptions in Lemma 2.5 hold, then within a finite number of iterations, only phase two is executed.*

Proof. First we claim that phase two must occur within a finite number of iterations. If on the contrary only phase one is occurred, then θ is decreased in each iteration, and will be decreased to $\theta < \theta^*$ after a finite number of iterations. Then according to Lemma 2.5, $\|g^{\mathcal{A}}(x^k)\| > \theta\|\nabla_\Omega f(x^k)\|$ will occur. Once this holds, phase one branches to phase two. This is a contradiction.

Once phase two is invoked, then phase two cannot branch to phase one infinite times. Otherwise, θ will be reduced to $\theta < \theta^*$ and again $\|g^{\mathcal{A}}(x^k)\| > \theta\|\nabla_\Omega f(x^k)\|$ will occur, and after that phase two cannot branch to phase one. ∎

Now we make clear the novelty of our new active set method in Algorithm 2.1, compared to the active set method proposed by Hager and Zhang [18]. Algorithm 2.1 adopts the so-called piecewise PG method with $x^{k+1} = P_\Omega[x^k - \alpha_k g(x^k)]$ so that the search direction within one iteration is along the projection arc [8]. While the active set method by Hager and Zhang [18] chooses the so-called gradient projection algorithm (GPA) in which the single projection is used to define the feasible search direction $d^k = P_\Omega[x^k - \bar{\alpha} g(x^k)] - x^k$ where $\bar{\alpha} > 0$ is a fixed parameter, and the next iterate point $x^{k+1} = x^k + s_k d^k$ is obtained by backtracking toward the starting point along d^k. As pointed out by Bertsekas in subsection 2.3 of [3] that the iterates obtained by the piecewise PG method used in this paper are more likely to be at the boundary than the GPA used in [18]. Moreover, the finite identification property of the new active set method is shown in Theorem 2.3. On contrast, after Lemma 6.1 of [18], the authors stated that there is a fundamental difference between the GPA and the PG method, and consequently they can not show the finite identification property of the active set method in [18].

The main motivation of such modification lies in that Algorithm 2.1 guarantees $\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$, which is novel and essential in providing the convergence result of the new smoothing active set method given in the next section. This convergence result is stronger than that of the active set method in [18] which guarantees $\liminf_{k\to\infty} \|d^1(x^k)\| = 0$, since by Lemma 2.2 of [8],

$$\|\nabla_\Omega f(x^k)\| = \lim_{\alpha\downarrow 0} \frac{\|P_\Omega[x^k - \alpha\nabla f(x^k)] - P_\Omega[x^k]\|}{\alpha} \geq \|d^1(x^k)\|. \tag{2.35}$$

But $\liminf_{k\to\infty} \|d^1(x^k)\| = 0$ does not imply $\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$ because the norm of projected gradient is not continuous and can be large near the solution. This can be explained by the following simple example.

Example 1 Let us consider the linearly constrained strongly convex quadratic programming

$$\begin{aligned} \min \quad & 0.01(10x_1 + x_2)^2 + 10(x_1 + 10.1x_2 + 1)^2 + x_3^2 \\ \text{s.t.} \quad & x_2 \geq 1,\ x_3 \geq 0. \end{aligned}$$

We know $\mathcal{M} = \mathcal{M}_I = \{1, 2\}$ for this problem. It is easy to calculate that $x^* = (x_1^*, x_2^*, x_3^*)^T = (-10.1, 1, 0)^T$ is the unique global minimizer. The Lagrangian multipliers corresponding to the constraint $x_2 \geq 1$ and $x_3 \geq 0$ at x^* are $\lambda_1^* = 200$ and

$\lambda_2^* = 0$, respectively. Hence $\mathcal{A}_+(x^*) = \{1\}$, $\mathcal{A}(x^*) = \mathcal{M} = \{1, 2\}$, and x^* is a degenerate stationary point. The tangent cone to the feasible region at x^*, and the gradient at x^* are

$$T(x^*) = \{(d_1, d_2, d_3)^T \in R^3 \ : \ d_2 \geq 0, \ d_3 \geq 0\}, \quad \nabla f(x_1^*, x_2^*, x_3^*) = (0, 200, 0)^T.$$

Let $x^k = (x_1^k, x_2^k, x_3^k)^T = (-10.1 + (0.5)^{k/2}, 1 + (0.5)^k, (0.5)^k)^T \to x^*$ as $k \to +\infty$. By direct computation, the tangent cone to the feasible region at x^k is $T(x^k) = R^3$. Since f is twice continuously differentiable near x^*, we know that

$$\nabla f(x^k) \to \nabla f(x^*) = (0, 200, 0)^T \quad \text{as } k \to \infty,$$

and consequently

$$\|\nabla_\Omega f(x^k)\| = \|P_{T(x^k)}[-\nabla f(x^k)]\| = \| - \nabla f(x^k)\| \to 200 \quad \text{as } k \to \infty.$$

Hence $\lim\limits_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 200 > 0$, although $\lim\limits_{k\to\infty} \|d^1(x^k)\| = 0$.

Remark 2.7. Suppose $\{x^k\} \to x^*$, ∇f is locally Lipschitz continuous at x^*, and the active constraints are identified after finite iterations. Then there exists $k_0 > 0$ such that $T(x^k) \equiv T(x^*)$ for all $k \geq k_0$, and consequently $\lim_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$ and $\lim_{k\to\infty} \|d^1(x^k)\| = 0$ are equivalent. However, the active set method in [18] may not identify the active constraints, but only owns the property in Lemma 6.2 of [18] that the violation of the constraints $c_i^T x - d_i = 0$ for $i \in \mathcal{A}_+(x^*)$ by iterate x^k is on the order of the error in x^k squared under certain conditions. Using Example 1, we find

$$\bar{x}^k = \text{argmin}_y\{\|x^k - y\| \ : \ y_2 = 1\} = (x_1^k, 1, x_3^k)^T,$$

and

$$\lim_{k\to\infty} \frac{\|x^k - \bar{x}^k\|}{\|x^k - x^*\|^2} = \lim_{k\to\infty} \frac{|x_2^k - 1|}{\|x^k - x^*\|^2} = \lim_{k\to\infty} \frac{(0.5)^k}{(0.5)^k + (0.5)^{2k} + (0.5)^{2k}} = 1.$$

This indicates that although under certain conditions any sequence generated by the active set method [18] satisfies the property in Lemma 6.2 of [18], this property does not guarantee $\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$ that we need in designing the smoothing active set method with convergence result.

3. Smoothing active set method. In this section, we develop a smoothing active set method for solving (1.1) with solid convergence result. Here the objective function f is continuous, but not necessarily Lipschitz continuous.

To characterize the stationary points of (1.1), we review first the concepts of rly several subdifferentials that are often used in nonsmooth analysis [6, 31] and references therein. Let $f : R^n \to R$ be a proper lower semi-continuous function and $x \in R^n$ be a point where $f(x)$ is finite. The Fréchet subdifferential, the limiting (or Mordukhovich) subdifferential, the horizontal (or singular Mordukhovich) subdifferential, and the Clarke subdifferential (Definition 1 of [6]) are defined respectively as

$$\begin{aligned}
\hat{\partial} f(x) &:= \{v \ : \ f(y) \geq f(x) + v^T(y - x) + o(\|y - x\|), \ \forall y\}, \\
\partial f(x) &:= \left\{v \ : \ \exists x^k \xrightarrow{f} x, v^k \to v \text{ with } v^k \in \hat{\partial} f(x^k), \ \forall k\right\}, \\
\partial^\infty f(x) &:= \left\{v \ : \ \exists x^k \xrightarrow{f} x, t_k v^k \to v, t_k \downarrow 0 \text{ with } v^k \in \hat{\partial} f(x^k), \ \forall k\right\}, \\
\partial^\circ f(x) &:= \bar{\text{co}}\{\partial f(x) + \partial^\infty f(x)\},
\end{aligned}$$

where $x^k \xrightarrow{f} x$ means that $x^k \to x$ and $f(x^k) \to f(x)$, and "$\bar{\text{co}}$" means the closure of convex hull. We say that x^* is a Clarke stationary point of (1.1), if there is $V \in \partial^\circ f(x^*)$ such that

$$\langle V, x^* - z \rangle \leq 0 \quad \text{for all } z \in \Omega. \tag{3.1}$$

If there exists $V \in \partial f(x^*)$ such that (3.1) holds, then x^* is a limiting stationary point of (1.1). Under the basic qualification (BQ)

$$-\partial^\infty f(x^*) \cap N_\Omega(x^*) = \{0\}, \tag{3.2}$$

if x^* is a local minimizer, then x^* is a limiting stationary point (Rockafellar and Wets, Theorem 8.15 of [31]). It is easy to see that BQ in (3.2) holds if f is locally Lipschitz continuous at x^*, or x^* is an interior point of Ω. However, BQ often fails if f is non-Lipschitz at a boundary point x^* as pointed out in [9].

We use the following definition for smoothing function.

Definition 3.1. *Let $f : R^n \to R$ be a continuous function. We call $\tilde{f} : R^n \times R_+ \to R$ a smoothing function of f, if $\tilde{f}(\cdot, \mu)$ is continuously differentiable in R^n for any $\mu \in R_{++}$, and for any $x \in R^n$,*

$$\lim_{z \to x,\ \mu \downarrow 0} \tilde{f}(z, \mu) = f(x), \tag{3.3}$$

and there exists a constant $\kappa > 0$ and a function $\omega : R_{++} \to R_{++}$ such that

$$|\tilde{f}(x, \mu) - f(x)| \leq \kappa\omega(\mu) \quad \textit{with} \quad \lim_{\mu \downarrow 0} \omega(\mu) = 0. \tag{3.4}$$

For each fixed $\mu > 0$, the smooth subproblem is then defined by

$$\min \ \tilde{f}(x, \mu) \quad \text{s.t.} \quad x \in \Omega, \tag{3.5}$$

and the projected gradient $\nabla_\Omega \tilde{f}(x, \mu)$ is defined by

$$\nabla_\Omega \tilde{f}(x, \mu) \equiv P_{T(x)}[-\nabla_x \tilde{f}(x, \mu)] = \text{argmin}\{\|v + \nabla_x \tilde{f}(x, \mu)\| \ : \ v \in T(x)\},$$

where $T(x)$ is the tangent cone to Ω at x. Now we present our smoothing active set method, Algorithm 3.1.

Algorithm 3.1 Smoothing active set method

1: Let $\hat{\gamma}$ be a positive constant, ζ be a constant in $(0, 1)$, and $n_1 > 0$ be a positive integer. Choose $x^0 \in \Omega$ and $\mu_0 > 0$.
For $k \geq 0$:
2: Let $y^{0,k} = x^k$, $j := 0$.
3: **while** $\|\nabla_\Omega \tilde{f}(y^{j,k}, \mu_k)\| > \hat{\gamma}\mu_k$ or $j < n_1$, **do**
4: Execute one iterate of the active set method in Algorithm 2.1 for (3.5) with $\mu = \mu_k$ from the initial point $y^{j,k}$ and get the new point $y^{j+1,k}$.
Set $j := j + 1$.
5: **end while**
6: Set $x^{k+1} = y^{j,k}$.
7: Choose $\mu_{k+1} \leq \zeta\mu_k$.

Remark 3.2. It is worth mentioning that Algorithm 3.1 can be extended to a general framework of smoothing method, since the new active set method in Algorithm 2.1 that used in Algorithm 3.1 can be substituted by any other type of algorithm for minimizing smooth function (SA for short) on a closed convex set, as long as the algorithm satisfies the SA Requirement defined below. And then the same convergence result developed in this section can be obtained without difficulty.

SA Requirement For any fixed $\mu > 0$, let $\{x^k\}$ be generated by the SA that solves (3.5). Then

$$\liminf_{k\to\infty} \nabla_\Omega \tilde{f}(x^k, \mu) = 0.$$

When $\Omega = R^n$, then (3.5) reduces to unconstrained smooth optimization and hence $\nabla_\Omega \tilde{f}(x, \mu) = -\nabla \tilde{f}(x, \mu)$. Many unconstrained algorithms (UAs) for (3.5) meet the SA Requirement, e.g., the steepest descent method, the accelerated gradient method proposed by Nesterov, the conjugate gradient method, the trust region method, and the quasi-Newton method. When Ω is a general closed convex set, the projected gradient method satisfies the SA Requirement. When Ω is constructed by linear constraints defined in (1.2), the new active set method developed in section 2 meets the SA Requirement as we desired. Although the proposed active set method is in spirit very similar to Hager and Zhang's approach [18], the satisfaction of the SA Requirement makes it necessary and novelty for building up the convergence of the smoothing active set method that tackles linearly constrained non-Lipschitz nonconvex optimization problems.

Since we use a smoothing function in Algorithm 3.1, the convergence result is natural to connect with the smoothing function employed.

DEFINITION 3.3. *We say that x^* is a stationary point of (1.1) associated with a smoothing function $\tilde{f}$, if*

$$\liminf_{x\to x^*,\ x\in\Omega,\ \mu\downarrow 0} \langle \nabla_x \tilde{f}(x, \mu), x - z\rangle \le 0 \quad \textit{for all } z \in \Omega. \tag{3.6}$$

For any fixed $x \in \Omega$, denote

$$G_{\tilde{f}}(x) := \{V \ : \ \exists N \in \mathcal{N}_\infty^\sharp,\ x^\nu \underset{N}{\to} x,\ \mu_\nu \downarrow 0 \quad \text{with} \quad \nabla_x \tilde{f}(x^\nu, \mu_\nu) \underset{N}{\to} V\}. \tag{3.7}$$

By Corollary 8.47 (b) in [31], we have

$$\partial f(x) \subseteq G_{\tilde{f}}(x).$$

When f is Lipschitz continuous, it is shown in [7, 10, 31] that many smoothing functions satisfy the gradient consistency property

$$\partial^\circ f(x^*) = G_{\tilde{f}}(x^*).$$

Then the stationary point of (1.1) associated with $\tilde{f}$ coincides to the Clarke stationary point, i.e., there exists $V \in \partial^\circ f(x^*)$ such that (3.1) holds. When f is continuously differentiable at x^*, then $\partial^\circ f(x^*) = \{\nabla f(x^*)\}$ and x^* coincides to the classic stationary point for smooth minimization problems.

Now we show that x^* being a stationary point of (1.1) associated with a smoothing function $\tilde{f}$ is a necessary optimality condition for x^* being a local minimizer, without the requirement for BQ.

PROPOSITION 3.4. *For any given smoothing function $\tilde{f}$ defined in Definition 3.1, if x^* is a local minimizer of (1.1), then x^* is a stationary point of (1.1) associated with $\tilde{f}$.*

Proof. Since x^* is a local minimizer of (1.1), there exists a constant $\delta > 0$ such that

$$f(x^*) \le f(x) \quad \text{for any } x \in B_\delta(x^*) \cap \Omega.$$

This, combined with (3.4) in Definition 3.1 for the smoothing function, yields that for all $x \in B_\delta(x^*) \cap \Omega$,

$$\tilde{f}(x^*, \mu) \le f(x^*) + \kappa\omega(\mu) \le f(x) + \kappa\omega(\mu) \le \tilde{f}(x, \mu) + 2\kappa\omega(\mu). \tag{3.8}$$

For any $z \in \Omega$, let $x_\mu = x^* + \sqrt{\omega(\mu)}(z - x^*)$. Since Ω is a convex set and $\lim_{\mu\downarrow 0} \omega(\mu) = 0$, we get $x_\mu \in B_\delta(x^*) \cap \Omega$ for all μ sufficiently small and $x_\mu \to x^*$ as $\mu \downarrow 0$. By Taylor's theorem,

$$\begin{aligned} \tilde{f}(x^*, \mu) &= \tilde{f}(x_\mu, \mu) + \nabla_x \tilde{f}(x_\mu, \mu)^T (x^* - x_\mu) + o(\|x^* - x_\mu\|) \\ &= \tilde{f}(x_\mu, \mu) + \sqrt{\omega(\mu)} \nabla_x \tilde{f}(x_\mu, \mu)^T (x^* - z) + o(\sqrt{\omega(\mu)}). \end{aligned} \tag{3.9}$$

Substituting (3.9) into the left side of (3.8) and replacing x by x_μ into the right side of (3.8), we get

$$\sqrt{\omega(\mu)} \nabla_x \tilde{f}(x_\mu, \mu)^T (x^* - z) + o(\sqrt{\omega(\mu)}) \le 2\kappa\omega(\mu).$$

Dividing both sides of the above inequality by $\sqrt{\omega(\mu)}$, and taking the limit as $\mu \downarrow 0$, we find

$$\limsup_{\mu\downarrow 0} \langle \nabla_x \tilde{f}(x_\mu, \mu), x^* - z \rangle \le 0. \tag{3.10}$$

Note that

$$\langle \nabla_x \tilde{f}(x_\mu, \mu), x_\mu - z \rangle = (1 - \sqrt{\omega(\mu)}) \langle \nabla_x \tilde{f}(x_\mu, \mu), x^* - z \rangle.$$

This, together with (3.10), yields that

$$\liminf_{\mu\downarrow 0} \langle \nabla_x \tilde{f}(x_\mu, \mu), x_\mu - z \rangle = \liminf_{\mu\downarrow 0} (1 - \sqrt{\omega(\mu)}) \langle \nabla_x \tilde{f}(x_\mu, \mu), x^* - z \rangle \le 0,$$

which indicates

$$\liminf_{x \to x^*,\ x\in\Omega,\ \mu\downarrow 0} \langle \nabla_x \tilde{f}(x, \mu), x - z \rangle \le 0 \quad \text{for all } z \in \Omega. \tag{3.11}$$

Hence (3.6) holds and x^* is a stationary point of (1.1) with respect to $\tilde{f}$. ∎

Now we are ready to give the global convergence result of Algorithm 3.1.

THEOREM 3.5. *Assume Assumption 2.1 holds. Then any accumulation point x^* of $\{x^k\}$ generated by Algorithm 3.1 is a stationary point of (1.1) associated with the smoothing function $\tilde{f}$.*

Proof. By (3.4) of Definition 3.1, for each fixed $\mu > 0$,

$$f(x) - \kappa\omega(\mu) \leq \tilde{f}(x,\mu) \leq f(x) + \kappa\omega(\mu).$$

Then for each fixed $\mu > 0$,

$$\mathcal{L}_{\mu,\Gamma} = \{x \in \Omega \ : \ \tilde{f}(x,\mu) \leq \Gamma\}$$

is bounded for any Γ, because $\tilde{f}(x,\mu) \leq \Gamma$ implies $f(x) \leq \Gamma + \kappa\omega(\mu)$ and $\mathcal{L}_{\Gamma+\kappa\omega(\mu)}$ is bounded by Assumption 2.1.

By (2.11) of Theorem 2.2, we know Algorithm 3.1 is well-defined and

$$\|\nabla_\Omega \tilde{f}(x^{k+1},\mu_k)\| \leq \hat{\gamma}\mu_k, \quad \text{and} \quad \lim_{k\to\infty} \mu_k = 0. \tag{3.12}$$

According to Calamai and Moré [8],

$$\min\{\langle \nabla_x \tilde{f}(x^{k+1},\mu_k), v\rangle \ : \ v \in T(x^{k+1}), \ \|v\| \leq 1\} = -\|\nabla_\Omega \tilde{f}(x^{k+1},\mu_k)\|. \tag{3.13}$$

For any $z \in \Omega$, it is easy to see that

$$v = \frac{z - x^{k+1}}{\|z - x^{k+1}\|} \in T(x^{k+1}) \quad \text{and} \quad \|v\| = 1,$$

and hence by (3.13)

$$\langle \nabla_x \tilde{f}(x^{k+1},\mu_k), x^{k+1} - z\rangle \leq \|\nabla_\Omega \tilde{f}(x^{k+1},\mu_k)\|\|z - x^{k+1}\|.$$

This, combined with (3.12), yields

$$\langle \nabla_x \tilde{f}(x^{k+1},\mu_k), x^{k+1} - z\rangle \leq \hat{\gamma}\mu_k\|z - x^{k+1}\| \quad \text{for any } z \in \Omega. \tag{3.14}$$

Since x^* is an accumulation point of $\{x^k\}$, there exists an infinite sequence $\hat{K} \in \mathcal{N}_\infty^\sharp$ such that $\lim_{k\to\infty,\ k\in\hat{K}} x^k = x^*$. Let us denote $K = \{k-1 \ : \ k \in \hat{K}\}$ and then $\lim_{k\to\infty,\ k\in K} x^{k+1} = x^*$. We get from (3.14) that

$$\liminf_{k\to\infty,\ k\in K} \langle \nabla_x \tilde{f}(x^{k+1},\mu_k), x^{k+1} - z\rangle \leq 0 \quad \text{for any } z \in \Omega. \tag{3.15}$$

Therefore x^* is a stationary point of (1.1) associated with $\tilde{f}$. ∎

The objective function f in this paper is a general non-Lipschitz nonconvex function, which is broader than that considered in [4, 5, 11, 26]. In [5], the optimality and complexity for the convexly-constrained minimization problem are considered with the objective function in the following form

$$f(x) := \Theta(x) + c(h(x)), \quad \text{with} \quad h(x) := (h_1(D_1^T x), h_2(D_2^T x), \ldots, h_m(D_m^T x))^T.$$

Here $\Theta : R^n \to R$ and $c : R^m \to R$ are continuously differentiable, $D_i \in R^{n\times r}$, and $h_i : R^r \to R$, $i = 1,\ldots,m$ are continuous, but not necessarily Lipschitz continuous. This type of functions include all the objective functions considered in [4, 11, 26]. A generalized stationary point based on the generalized directional derivative is proposed in Definition 2 of [5], which is shown to be a necessary optimality condition, and satisfies the necessary optimality conditions given or used in [4, 11, 26]. Note that any $v \in T(x^{k+1})$ and $\|v\| \le 1$, there exists $z \in \Omega$ such that $v = z - x^{k+1} \in T(x^{k+1})$. By (3.14) of Theorem 3.5 and $\|z - x^{k+1}\| \le 1$,

$$\langle \nabla_x \tilde{f}(x^{k+1}, \mu_k), v\rangle = \langle \nabla_x \tilde{f}(x^{k+1}, \mu_k), z - x^{k+1}\rangle \ge -\hat{\gamma}\mu_k \|z - x^{k+1}\| \ge -\hat{\gamma}\mu_k,$$

which implies that (44) in Corollary 2 of [5] holds, and consequently any accumulation point of $\{x^k\}$ generated by the smoothing active set method is also a generalized stationary point of (1.1) defined in [5] for the same type of functions in [5] and Ω defined in (1.2).

Remark 3.6. In Algorithm 3.1, we require for each fixed μ_k, the iterations of the inner loop is no less than n_1. This strategy has no effect for convergence analysis, but aims to enhance the computational performance of finding a better stationary point with respect to $\tilde{f}$.

3.1. $\ell_2 - \ell_p$ sparse optimization model. Problem (1.3) is a special case of problem (1.1), for which we show that Algorithm 3.1 has stronger convergence results than that in Theorem 3.5.

For $|t|$, we construct its smoothing function as follows,

$$s_\mu(t) = \begin{cases} |t| & \text{if } |t| \ge \mu, \\ \frac{t^2}{2\mu} + \frac{\mu}{2} & \text{if } |t| < \mu. \end{cases} \tag{3.16}$$

By simple computation, for any $p \in (0,1)$ and any $t \in R$, we have $|s_\mu(t)^p - |t|^p| \le 2\mu^p$. We then easily find that

$$\tilde{f}(x, \mu) = \|Ax - b\|^2 + \tau \sum_{i=1}^{n} (s_\mu(x_i))^p$$

is a smoothing function of the objective function f in (1.3), and for any $x \in R^n$,

$$|\tilde{f}(x, \mu) - f(x)| \le \kappa\mu^p, \quad \text{with } \kappa = 2\tau n. \tag{3.17}$$

The gradient of $\tilde{f}(x, \mu)$ is

$$\nabla_x \tilde{f}(x, \mu) = 2A^T(Ax - b) + \tau p \sum_{i=1}^{n} (s_\mu(x_i))^{p-1} s'_\mu(x_i). \tag{3.18}$$

THEOREM 3.7. *There exists at least one accumulation point x^* of $\{x^k\}$ generated by Algorithm 3.1 with the smoothing function $\tilde{f}$. Suppose $\lim_{k\to\infty,\ k\in K} x^{k+1} = x^*$. Then $\{\lim_{k\to\infty,\ k\in K} \nabla_x \tilde{f}(x^{k+1}, \mu_k)\}$ is nonempty and bounded, and x^* is a limiting stationary point of (1.3).*

Proof. Assumption 2.1 holds for f in (1.3), since the objective function in (1.3) satisfies that $f(x) \to +\infty$ if $\|x\| \to +\infty$. Moreover, we know from (3.17) that

$$\tilde{f}(x^{j+1}, \mu_j) - f(x^{j+1}) \ge -\kappa\mu_j^p \quad \text{and} \quad \tilde{f}(x^j, \mu_j) - f(x^j) \le \kappa\mu_j^p.$$

Therefore for any natural number k,

$$\begin{aligned} f(x^{k+1}) &\le \tilde{f}(x^{k+1}, \mu_k) + \kappa\mu_k^p \le \tilde{f}(x^k, \mu_k) + \kappa\mu_k^p \le f(x^k) + 2\kappa\mu_k^p \\ &\le \cdots \\ &\le f(x^0) + 2\kappa[\mu_0^p + (\zeta\mu_0)^p + (\zeta^2\mu_0)^p + \ldots + (\zeta^k\mu_0)^p] \\ &\le f(x^0) + 2\kappa\mu_0^p \frac{1}{1 - \zeta^p}. \end{aligned}$$

Hence $\{x^k\}$ is bounded and there exists at least one accumulation point x^* of $\{x^k\}$ generated by Algorithm 3.1.

For any index i_0 such that $x^*_{i_0} > 0$, by direct computation,

$$\lim_{k\to\infty,\ k\in K} (\nabla_x \tilde{f}(x^{k+1}, \mu_k))_{i_0} = (2A^T(Ax^* - b))_{i_0} + \tau p(x^*_{i_0})^{p-1}.$$

For i_0 such that $x^*_{i_0} = 0$, let $K_2 = \{k \in K \ :\ x^{k+1}_{i_0} > 0\}$. If K_2 is an infinite subsequence, then we define $z^{k+1,1}$ and $z^{k+1,2}$ in R^n_+ for each $k \in K_2$, where

$$z^{k+1,1}_i = \begin{cases} x^{k+1}_i & \text{if } i \neq i_0, \\ 0 & \text{if } i = i_0, \end{cases} \quad \text{and} \quad z^{k+1,2}_i = \begin{cases} x^{k+1}_i & \text{if } i \neq i_0, \\ 2x^{k+1}_i & \text{if } i = i_0. \end{cases}$$

Replacing $z^{k+1,1}$ and $z^{k+1,2}$ in (3.14) of Theorem 3.5 respectively, we get eventually

$$-\hat{\gamma}\mu_k \le (\nabla_x \tilde{f}(x^{k+1}, \mu_k))_{i_0} \le \hat{\gamma}\mu_k \quad \text{for any} \quad k \in K_2,$$

and consequently

$$\lim_{k\to\infty,\ k\in K_2} (\nabla_x \tilde{f}(x^{k+1}, \mu_k))_{i_0} = 0. \tag{3.19}$$

Otherwise, there exists an integer $\bar{k} > 0$ such that $x^{k+1}_{i_0} = 0$ for all $k \ge \bar{k}$, $k \in K$. In this case

$$\begin{aligned}
(\nabla_x \tilde{f}(x^{k+1}, \mu_k))_{i_0} &= (2A^T(Ax^{k+1} - b))_{i_0} + \tau p(s_{\mu_k}(x^{k+1}_{i_0}))^{p-1} s'_{\mu_k}(x^{k+1}_{i_0}) \\
&= (2A^T(Ax^{k+1} - b))_{i_0} + \tau p(\frac{\mu_k}{2})^{p-1} \frac{x^{k+1}_{i_0}}{\mu_k} \\
&= (2A^T(Ax^{k+1} - b))_{i_0} \qquad \text{for all } k \ge \bar{k},\ k \in K.
\end{aligned}$$

Consequently

$$\lim_{k\to\infty,\ k\in K} (\nabla_x \tilde{f}(x^{k+1}, \mu_k))_{i_0} = (2A^T(Ax^* - b))_{i_0}. \tag{3.20}$$

Combining (3.19) and (3.20), we can easily find that any accumulation point $V \in R^n$ of $\{\nabla_x \tilde{f}(x^{k+1}, \mu_k)\}_K$ is of the special form

$$V_i = \begin{cases} (2A^T(Ax^* - b))_i + \tau p(x^*_i)^{p-1} & \text{if } x^*_i > 0 \\ (2A^T(Ax^* - b))_i \ \text{ or } \ 0, & \text{if } x^*_i = 0 \end{cases} \tag{3.21}$$

that is bounded.

Furthermore, we know $V \in \partial f(x)$ by the definition of the limiting subdifferential, which indicates that x^* is also a limiting stationary point of (1.3). ∎

THEOREM 3.8. *Let x^* be an accumulation point of a sequence $\{x^k\}$ generated by Algorithm 3.1 for solving (1.3). If $\mathcal{F}(x^*) = \emptyset$, then $x^* = 0$ is a local minimizer of (1.3). If $\mathcal{F}(x^*) \neq \emptyset$ and*

$$2(A^TA)_{\mathcal{F}(x^*)\mathcal{F}(x^*)} + \tau p(p-1)\mathrm{diag}((x^*_{\mathcal{F}(x^*)})^{p-2}) \quad \textit{is positive definite,} \tag{3.22}$$

then x^* is a strict local minimizer of (1.3).

Proof. By Theorem 3.7, and (3.15) in the proof of Theorem 3.5, there exists an accumulation point V of $\{\lim_{k\to\infty,\ k\in K} \nabla_x \tilde{f}(x^{k+1},\mu_k)\}$ in the form of (3.21) such that

$$\langle V, x^* - z\rangle \le 0 \quad \text{for all } z \ge 0.$$

This indicates $V_i = 0$ for all $i \in \mathcal{F}(x^*)$.

Let us define $\varsigma_i := \frac{2}{\tau}\Big(\max\{-(A^T(Ax^*-b))_i, 0\} + 1\Big)$ for all $i \in \mathcal{A}(x^*)$, and

$$\bar{f}(x) := \|Ax-b\|^2 + \tau \sum_{i\in\mathcal{F}(x^*)} |x_i|^p + \tau \sum_{i\in\mathcal{A}(x^*)} \varsigma_i x_i. \tag{3.23}$$

Now we consider the minimization problem

$$\min\ \bar{f}(x) \quad \text{s.t.} \quad x \ge 0, \tag{3.24}$$

whose objective function is twice continuously differentiable around $x^* \in R^n_+$. By direct computation, $\bar{f}(x^*) = f(x^*)$ and the gradient $\nabla \bar{f}(x^*)$ has the form

$$(\nabla \bar{f}(x^*))_i = \begin{cases} (2A^T(Ax^*-b))_i + \tau p (x_i^*)^{p-1} & \text{if } i \in \mathcal{F}(x^*), \\ (2A^T(Ax^*-b))_i + \tau \varsigma_i & \text{if } i \in \mathcal{A}(x^*). \end{cases}$$

Clearly, $(\nabla \bar{f}(x^*))_i = V_i = 0$ for all $i \in \mathcal{F}(x^*)$ and $(\nabla \bar{f}(x^*))_i \ge 2$ for all $i \in \mathcal{A}(x^*)$. Therefore, x^* is a stationary point of (3.24) since

$$x^* \ge 0,\ \nabla \bar{f}(x^*) \ge 0,\ {x^*}^T \nabla \bar{f}(x^*) = 0. \tag{3.25}$$

Note that for any $p \in (0,1)$,

$$\lim_{t\downarrow 0,\ t\neq 0} \frac{t^p}{t} = \lim_{t\downarrow 0,\ t\neq 0} t^{p-1} = +\infty.$$

Thus there exists $\delta_1 > 0$ such that for any $x \in B_{\delta_1}(x^*) \cap R^n_+$

$$\varsigma_i x_i \le x_i^p \quad \text{for all } \ i \in \mathcal{A}(x^*).$$

Consequently for any $x \in B_{\delta_1}(x^*) \cap R^n_+$,

$$\bar{f}(x) - f(x) = \tau \sum_{i\in\mathcal{A}(x^*)} (\varsigma_i x_i - x_i^p) \le 0. \tag{3.26}$$

If $\mathcal{F}(x^*) = \emptyset$, then $x^* = 0$ and $\bar{f}(x)$ in (3.23) is a convex function. Any stationary point of (3.24) is a global minimizer of (3.24). Hence

$$\bar{f}(x^*) \le \bar{f}(x) \quad \text{for any } x \in R^n_+.$$

This, combined with (3.26), yields

$$f(x^*) = \bar{f}(x^*) \le \bar{f}(x) \le f(x) \quad \text{for any } x \in B_{\delta_1}(x^*) \cap R^n_+.$$

Hence x^* is a local minimizer of (1.3).

Now we consider $\mathcal{F}(x^*) \neq \emptyset$. Noting (3.25), we know that (x^*, λ^*) satisfies the KKT conditions if and only if $\lambda^* = \nabla \bar{f}(x^*)$. Since for any $i \in \mathcal{A}(x^*)$, $\lambda_i^* = (\nabla \bar{f}(x^*))_i \ge 2$, it follows that the critical cone

$$\mathcal{C}(x^*, \lambda^*) = \{d \in R^n \ :\ d_i = 0 \text{ for } i \in \mathcal{A}(x^*),\ \text{and } d_i \ge 0 \text{ for } i \in \mathcal{F}(x^*)\ \}.$$

It is easy to see that (3.22) is equivalent to

$$d^T \nabla^2 \bar{f}(x^*) d > 0 \quad \text{for any } d \in \mathcal{C}(x^*, \lambda^*),\ d \neq 0,$$

which are the second-order sufficient conditions for x^* being a strict local minimizer of (3.24). Then there exists $\delta > 0$ such that

$$f(x^*) = \bar{f}(x^*) < \bar{f}(x) \quad \text{for any } x \in B_\delta(x^*) \cap R^n_+. \tag{3.27}$$

This, combined with (3.26), yields

$$f(x^*) < f(x) \quad \text{for any } x \in B_{\breve{\delta}}(x^*) \cap R^n_+,$$

where $\breve{\delta} = \min\{\delta, \delta_1\}$. Hence x^* is a strict local minimizer of (1.3). ∎

4. Numerical experiments. Hyperspectral image is a 3D image cube at hundreds of contiguous and narrow spectral channels often used in earth observation and remote sensing. Due to the low spatial resolution of hyperspectral cameras, pixels are often a mixture of several spectra of materials in a scene. This, together with the 3D image cube, makes the hyperspectral image hard to display and understand. Hyperspectral unmixing is the process of estimating a common set of spectral bases (called endmembers) and their corresponding composite percentages (called abundance) at each pixel so that people can better visualize, analyze and understand the hyperspectral image.

In this section, we apply Algorithm 3.1 with Algorithm 2.1 to the constrained sparse nonnegative matrix factorization (NMF) used in hyperspectral unmixing. The mathematical model is as follows.

$$\min_{W,H} \quad \frac{1}{2}\|V - WH\|_F^2 + \tau\|H\|_p^p \tag{4.1}$$

$$\text{s.t.} \quad W \geq 0,\ H \geq 0, \tag{4.2}$$

$$1_K^T H = 1_N^T, \tag{4.3}$$

where $V = [v_1, v_2, \ldots, v_N] \in R_+^{L\times N}$ is the given hyperspectral image data with L channels and N pixels, $W = [w_1, w_2, \ldots, w_K] \in R_+^{L\times K}$ is the endmember matrix including K endmember vectors with $K \ll \min\{L, N\}$, and $H = [h_1, h_2, \ldots, h_N] \in R_+^{K\times N}$ is the corresponding abundance matrix. Here 1_K and 1_N are the column vectors of all ones of dimension K and N, respectively.

In the objective function in (4.1), the parameter $\tau > 0$ balances the data fidelity term $\frac{1}{2}\|V - WH\|_F^2$ and the sparse regularization term $\|H\|_p^p$, $p \in (0, 1)$ that forces the sparsity of the abundance matrix. The sparse regularization term is effective for spectral unmixing since only a few endmembers can contribute to representing an observed pixel. To be physically meaningful, the nonnegative constraints in (4.2) are necessary. Moreover, the abundance sum-to-one constraints (ASC) in (4.3) are required since each column of H is the abundance vector whose components are the proportions of each endmember contributing to the mixed pixel. Let H_{ij} denote the (i, j)-entry of the matrix H. The existence of ASC makes the usually used sparsity-induced regularization term $\|H\|_1 = \sum_{i,j} |H_{ij}|$ meaningless since in this case $\|H\|_1$ equals a constant N.

To solve the constrained sparse NMF model, the two block coordinate descent method is adopted. That is, W and H are considered to be two separate block variables, and the scheme alternatively solves the two subproblems of matrix-based optimization problems. The difficulty of solving problem (4.1)-(4.3) for block H lies in two aspects: the non-Lipschitz regularization term of the objective function in (4.1) and the numerous N constraints defined by ASC in (4.3).

In [30], Qian et al. considered the special case $p = \frac{1}{2}$ and called the model $L_{1/2}$-NMF. To deal with the ASC, Qian et al. adopted the strategy akin to that in [20] by augmenting the data matrix V and the endmember matrix W to V_a and W_a as

$$V_a = \begin{pmatrix} V \\ \delta 1_N^T \end{pmatrix} \quad \text{and} \quad W_a = \begin{pmatrix} W \\ \delta 1_K^T \end{pmatrix}, \tag{4.4}$$

where $\delta > 0$ controls the impact of the additivity constraint over the endmember abundances. This strategy, in fact, leads to solve the penalized counterpart

$$\min_{W\geq 0,\ H\geq 0} \quad \frac{1}{2}\|V - WH\|_F^2 + \tau\|H\|_p^p + \frac{1}{2}\delta^2\|1_K^T H - 1_N^T\|_F^2. \tag{4.5}$$

The multiplicative update (MU) method [23] for classic NMF is extended to solve the $L_{1/2}$-NMF, by alternatively updating W and H as

$$W \leftarrow W. * (VH^T)./(WHH^T), \tag{4.6}$$

$$H \leftarrow H. * (W_a^T V_a)./(W_a^T W_a H + \frac{\tau}{2} T_\xi(H)^{-\frac{1}{2}}), \tag{4.7}$$

where $(T_\xi(H)^{-\frac{1}{2}})_{ij} = H_{ij}^{-\frac{1}{2}}$ if $H_{ij} > \xi$ and $(T_\xi(H)^{-\frac{1}{2}})_{ij} = 0$ otherwise for a predefined threshold $\xi > 0$ to avoid computationally instability. Here ".∗" and "./" denote the elementwise matrix multiplication and division, respectively.

Here we use the two block proximal alternating optimization (PAO) framework to solve (4.5). Let W_a^k be the augmented matrix in (4.4) where the block W in W_a is replaced by W^k.

Algorithm 4.1 PAO Framework

1: Initialize $W^1 \geq 0$, $H^1 \geq 0$, and parameters $\tau_1 > 0$ and $\tau_2 > 0$.
2: Repeat until a stopping criterion is satisfied
   2.1 Find W^{k+1} and H^{k+1} such that

$$W^{k+1} = \arg\min_{W\geq 0}\{\frac{1}{2}\|V - WH^k\|_F^2 + \frac{1}{2}\tau_1\|W - W^k\|_F^2\}, \tag{4.8}$$

$$H^{k+1} = \arg\min_{H\geq 0}\{\frac{1}{2}\|V_a - W_a^k H\|_F^2 + \tau\|H\|_p^p + \frac{1}{2}\tau_2\|H - H^k\|_F^2\}. \tag{4.9}$$

   2.2 Set $k := k + 1$.

We combine Algorithm 2.1 and Algorithm 3.1 proposed in this paper to solve the two subproblems (4.8) and (4.9) in Algorithm 4.1.

- To solve the W-subproblem in (4.8), we use ASCG, i.e., Algorithm 2.1 with the LCO employing the conjugate gradient (CG) method [12].
- To solve the H-subproblem in (4.9) that involves the non-Lipschitz term, we use SASCG, i.e., Algorithm 3.1 with ASCG that solves the smoothing H-subproblem of (4.9). The smoothing function of $\|H\|_p^p$ is constructed by using (3.16).

We denote the method as PAO-ASCG-SASCG for short.

We also use the two block proximal alternating optimization (PAO) framework to solve (4.1)-(4.3) directly without penalization to the equality constraints, by substituting (4.9) in Algorithm 4.1 by

$$H^{k+1} = \arg \min_{H\geq 0,\ 1_K^T H = 1_N^T} \{F_{W^k,H^k}(H)\}, \tag{4.10}$$

where

$$F_{W^k,H^k}(H) := \frac{1}{2}\|V - W^k H\|_F^2 + \tau\|H\|_p^p + \frac{1}{2}\tau_2\|H - H^k\|_F^2. \tag{4.11}$$

We then combine Algorithm 2.1 and Algorithm 3.1 proposed in this paper to solve (4.8) and (4.10) in the PAO framework.

- To solve the W-subproblem in (4.8), we use the projected gradient method.
- To solve the H-subproblem in (4.10), we use SASPG, i.e., Algorithm 3.1, together with Algorithm 2.1 in which the LCO being the projected gradient method. The smoothing function of $\|H\|_p^p$ is also constructed by using (3.16).

We denote the method as PAO-PG-SASPG-O for short. Here '-O' indicates that the original $L_{1/2}$-NMF problem (4.1)-(4.3) is solved.

It is worth mentioning that the constraints in (4.10) are N independent simplex $h_j \geq 0$, $\sum_{i=1}^K H_{ij} = 1$, $j = 1, 2, \dots, N$. Let

$$\mathcal{A}(H^k) := \{(i,j) \ :\ H_{ij}^k = 0\},$$
$$\breve{\Omega}(H^k) := \{H \in \Omega \ :\ H_{ij} = 0 \quad \text{if } (i,j) \in \mathcal{A}(H^k)\}.$$

The efficiency of Algorithm 2.1 depends on the fast computation of matrices $P_\Omega[H]$, $P_{\breve{\Omega}(H^k)}[H]$, $\nabla_\Omega F_{W^k,H^k}(H)$, and $g^{\mathcal{A}}(H)$. Here $P_{\breve{\Omega}(H^k)}[H]$ is used for the projected gradient method that works on the faces $\breve{\Omega}(H^k)$ of Ω. All the four types of matrices are essentially composed by projections of a vector on a certain polyhedron. The projections of a vector on a polyhedron can be obtained efficiently, e.g., [18]. Here we compute them in matrix form directly, since N is in general no less than 10000. We adopt the Matlab code **SimplexProj** in [34] for obtaining $P_\Omega[H]$. And by using the grouping idea of inactive indices as in [22], we use **SimplexProj** for computing $P_{\breve{\Omega}(H^k)}[H]$ on each group with the same inactive constraints. Moreover, the projected gradient $\nabla_\Omega F_{W^k,H^k}(H)$, and $g^{\mathcal{A}}(H)$ can be computed efficiently in matrix form using the KKT conditions.

We use two real-world data in the experiment.

Jasper Ridge, is a popular hyperspectral data. There are 512×614 pixels in it. In this image, each pixel is recorded at 224 channels ranging from 0.38 to $2.5\mu m$, and the spectral resolution is up to $9.46nm$. Because this hyperspectral image is too complex to get the groundtruth, we consider a subimage of 100×100 as in [43], the first pixel of which is the $(105, 269)$-th pixel in the original image. After the channels 1–3, 108–112, 154–166 and 220–224 are removed (due to dense water vapor and atmospheric effects), we remain 198 channels (this is a common preprocess for hyperspectral unmixing analysis). There are 4 endmembers in groundtruth: ♯1 Tree, ♯2 Soil, ♯3 Water, ♯4 Road.

Urban, is one of the most widely used hyperspectral data in the hyperspectral unmixing study. There are 307×307 pixels, each of which corresponds to a 2×2

m^2 area. In this image, there are 210 wavelengths ranging from 400 nm to 2500 nm, resulting in a spectral resolution of 10 nm. After the channels 1–4, 76, 87, 101–111, 136–153 and 198–210 are removed, we remain 162 channels. There are 4 endmembers in ground truth: ♯1 Asphalt, ♯2 Grass, ♯3 Tree, ♯4 Roof.

We choose $p = \frac{1}{2}$ and consider the $L_{1/2}$-NMF problem. We compare our methods (PAO-ASCG-SASCG and PAO-PG-SASPG-O) with the other three methods. The information of all the methods are summarized as follows.

1) Our method: PAO-ASCG-SASCG that solves the penalized counterpart of $L_{1/2}$-NMF problem in (4.5).
2) Our method: PAO-PG-SASPG-O that solves the original $L_{1/2}$-NMF problem in (4.1)-(4.3).
3) PAO-PG-SPG-O: this method solves the original $L_{1/2}$-NMF problem in (4.1)-(4.3). It employs the PAO framework in Algorithm 4.1 with (4.9) substituted by (4.10). The W-subproblem is solved by the PG method [24] and the H-subproblem is solved by the smoothing projected gradient method [41]. No active set strategy is adopted.
4) MU method: this method is a state-of-art method that employs (4.6) and (4.7) recursively to solve the penalized counterpart of $L_{1/2}$-NMF problem in (4.5).
5) Adaptive HT method: this method is proposed in [35]. It employs the half-thresholding algorithm and an adaptive strategy for automatically choosing regularization parameters τ_j^k, $j = 1, 2, \ldots, N$ in kth iteration, and solving the penalized $L_{1/2}$ sparsity-constrained NMF defined by

$$\min_{W\geq 0, H\geq 0} \frac{1}{2}\|V - WH\|_F^2 + \sum_{j=1}^{N} \tau_j^k \|h_j\|_{\frac{1}{2}}^{\frac{1}{2}}. \tag{4.12}$$

We set the maximum CPU time to be 3000 seconds for all the methods, and the maximum number of iterations for the MU method to be 3000, and the maximum number of iterations for the PAO-ASCG-SASCG, PAO-PG-SASPG-O, and PAO-PG-SPG-O methods to be 1000, and $n_1 = 5$ in Algorithm 3.1. To overcome the nonconvexity of the original problem (4.1)-(4.3), and the penalized problem (4.5), we randomly choose 10 initial points for W^1 and H^1 using the Matlab commands rand(L, K) and rand(K, N) for all the methods, respectively. And each column of H^1 is further rescaled to be sum to one, according to the ASC in (4.3). The MU and the PAO-ASCG-SASCG methods involve two essential parameters τ and δ, while the Adaptive HT method only has one parameter δ, and the PAO-PG-SASCG-O methods only has one parameter τ. In order to estimate an optimal parameter, we first determine the intervals $[\tau_{\min}, \tau_{\max}]$, and/or $[\delta_{\min}, \delta_{\max}]$ by trying the values at large steps. We then search the optimal parameters by trying more values in the interval $[\tau_{\min}, \tau_{\max}]$, and/or $[\delta_{\min}, \delta_{\max}]$.

If (W, H) is a solution of NMF, then $(WD, D^{-1}H)$ is also a solution of NMF for any positive diagonal matrices D. To get rid of this kind of uncertainty, one intuitive method is to scale each column of W to be the unit ℓ_1- or ℓ_2-norm [39, 43], e.g.,

$$W_{lk} \leftarrow \frac{W_{lk}}{\sqrt{\sum W_{lk}^2}}, \quad H_{kn} \leftarrow H_{kn}\sqrt{\sum W_{lk}^2}. \tag{4.13}$$

Considering the ASC in (4.3), we further let

$$H_{kn} \leftarrow \frac{H_{kn}}{\sum_k H_{kn}}. \tag{4.14}$$

To evaluate the performance of the computed solution, we use the spectral angle distance (SAD) and the root mean squared error (RMSE) [30, 35, 43] as two benchmark metrics. The SAD is used to evaluate the endmembers, which is defined as

$$\mathrm{SAD}(w, \hat{w}) = \arccos\left(\frac{w^T \hat{w}}{\|w\|\|\hat{w}\|}\right), \tag{4.15}$$

where w is an estimated endmember, and $\hat{w}$ is the corresponding ground-truth endmember. The RMSE is used to evaluate the performance of the estimated abundance, which is given by

$$\mathrm{RMSE}(z, \hat{z}) = \left(\frac{1}{N}\|z - \hat{z}\|^2\right)^{1/2}, \tag{4.16}$$

where N is the number of pixels in the image, z is the estimated abundance map (a row vector in the abundance matrix H), and $\hat{z}$ is the corresponding ground-truth abundance map. In general, a smaller SAD and a smaller RMSE correspond to a better hyperspectral unmixing result.

We draw in Fig. 1 the corresponding objective value $\frac{1}{2}\|V - WH\|_F^2 + \tau\|H\|_{\frac{1}{2}}^{\frac{1}{2}}$ of each iterate point versus the CPU time obtained by the PAO-PG-SASPG-O and the PAO-PG-SPG-O method, using the same optimal parameter $\tau = 1.5 \times 10^6$, and the same initial point on Jasper Ridge data, respectively. We divide the x-axis to be $[0, 200]$ and $[200, 3000]$ in two subfigures to see clear the decrease tendency and the final objective value. We can find from Fig. 1 that our PAO-PG-SASPG-O decreases faster and gets lower objective value than the PAO-PG-SPG-O method. The final objective value obtained by the PAO-PG-SASPG-O method is 2.6494e10, which is much lower than 2.6988e10 that obtained by the PAO-PG-SPG-O method. It is easy to see that the active set strategy helps fasten the computational speed.

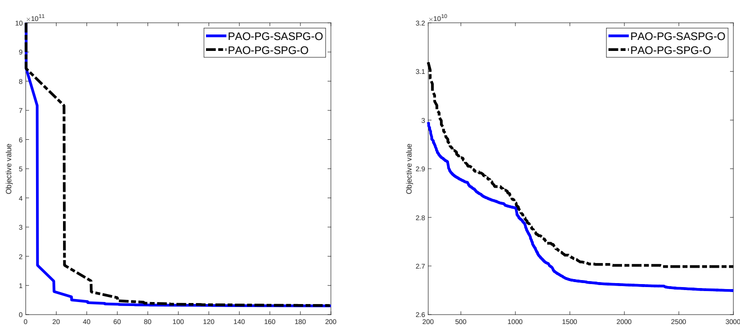

Fig. 1. *Convergence curve of objective value versus CPU time using the PAO-PG-SPG-O and the PAO-PG-SASPG-O on the Jasper Ridge data, respectively.*

For Jasper Ridge, we record in Table 1 the final SAD and RMSE for each endmember corresponding to the computed solution with the smallest sum of SAD and

RMSE, among the 10 trials of initial points as well as the choices of parameters. The lowest SAD and RMSE for each endmember, and the lowest average SAD and RMSE are indicated in bold face in Table 1. It is easy to see that the computed solution obtained by the PAO-PG-SASPG-O method proposed in this paper has the lowest average SAD and RMSE of the four endmembers. Our proposed PAO-ASCG-SASCG method that solves the penalized version of $L_{1/2}$-NMF also provides lower average SAD and RMSE than the MU and the Adaptive HT methods.

For Urban, we record in Table 2 the final SAD and the final RMSE for each endmember. The lowest SAD and RMSE for each endmember, and the lowest average SAD and RMSE are indicated in bold face in Table 2. Clearly the PAO-ASCG-SASCG method provides the solution that obtains the lowest average SAD and RMSE than the other four methods. The PAO-PG-SPG-O and PAO-ASCG-SASCG-O method for solving the original model (4.1)-(4.3) do not provide satisfying SAD and RMSE. The reason, we think, is due to the model itself. As pointed out in [43], applying an identical strength of constraints to all the factors, (that is, in our case, using the same $p = \frac{1}{2}$ for all the columns of H) does not hold in practice. Therefore, in [43] they proposed to solve

$$\min_{W\geq 0, H\geq 0} \frac{1}{2}\|V - WH\|^2 + \tau \sum_{j=1}^{N} \|h_j\|_{p_j}^{p_j}, \tag{4.17}$$

where $p_j \in (0, 1)$, $j = 1, 2, \ldots, N$, are estimated from the original data V using two-steps procedures. If the pixels indeed have very different levels of sparsity as in Urban, the sum-to-one constraints will make the original model (4.1)-(4.3) deviate a lot from the true model. The PAO-ASCG-SASCG method, in contrast, because of the lack of the sum-to-one constraints, has the ability to adjust the sparsity levels of different pixels to some degree. The Adaptive HT method, which adaptively adjusts the different regularization parameter for each column of H, also has the effect to assign different level of sparsity for each pixel. When the pixels have not so much different levels of sparsity as in Jasper, the PAO-PG-SASPG-O that solves the original model (4.1)-(4.3) with the sum-to-one constraints provides the best SAD and RMSE.

TABLE 1
SAD and RMSE on the Jasper Ridge data estimated by our methods and the other methods

	SAD				Avg.
Jasper Ridge ($K = 4$)	♯1	♯2	♯3	♯4	♯1 ∼ ♯4
MU	0.2070	0.1185	0.3324	0.2939	0.2379
Adaptive HT	0.1451	0.3099	0.1367	0.1515	0.1858
PAO-ASCG-SASCG	**0.1241**	0.0690	0.1859	0.1645	0.1359
PAO-PG-SPG-O	0.1315	**0.0606**	0.1132	**0.0516**	0.0892
PAO-PG-SASPG-O	0.1301	0.0616	**0.1019**	0.0609	**0.0886**
	RMSE				Avg.
MU	0.1235	0.0953	0.1773	0.0953	0.1361
Adaptive HT	0.1016	0.1483	0.1761	0.1885	0.1536
PAO-ASCG-SASCG	**0.0836**	**0.0425**	0.1244	0.1052	0.0889
PAO-PG-SPG-O	0.0846	0.0581	**0.0929**	0.0875	0.0808
PAO-PG-SASPG-O	0.0840	0.0578	0.0930	**0.0842**	**0.0798**

The abundance fractions for Jasper Ridge from the ground-truth, and separated by the five methods are shown in Fig. 2. We can also see that our proposed PAO-

TABLE 2
SAD and RMSE on the Urban data estimated by our methods and the other methods

	SAD				Avg.
Urban ($K = 4$)	$\sharp 1$	$\sharp 2$	$\sharp 3$	$\sharp 4$	$\sharp 1 \sim \sharp 4$
MU	0.1976	**0.0318**	0.0454	0.1445	0.1048
Adaptive HT	**0.0715**	0.0393	0.0704	0.3288	0.1275
PAO-ASCG-SASCG	0.0738	0.0525	**0.0314**	**0.0736**	**0.0578**
PAO-PG-SPG-O	0.0900	0.1940	0.0423	0.3424	0.1672
PAO-PG-SASPG-O	0.0925	0.1026	0.0397	0.2153	0.1125
	RMSE				Avg.
MU	**0.0989**	0.1037	0.0707	0.0995	0.0932
Adaptive HT	0.1165	**0.0964**	0.0794	0.0895	0.0954
PAO-ASCG-SASCG	0.1101	0.1085	**0.0562**	**0.0548**	**0.0824**
PAO-PG-SPG-O	0.2595	0.2242	0.1281	0.2052	0.2242
PAO-PG-SASPG-O	0.2452	0.1715	0.1435	0.2082	0.1921

ASCG-SASCG and PAO-PG-SASPG-O methods provide good estimates of abundance. The abundance fractions for Urban from the ground-truth, and separated by the MU, the Adaptive HT, and the PAO-ASCG-SASCG methods are shown in Fig. 3. It is easy to see that our proposed PAO-ASCG-SASCG method provide the best estimates of abundance.

The numerical results demonstrate that our proposed PAO-PG-SASPG-O method and PAO-ASCG-SASCG method can efficiently solve the original and penalized $L_{1/2}$-NMF problem, respectively. Moreover, at least one of our methods provides an excellent unmixing performance, compared to the popular MU method and the Adaptive HT method.

It is worth pointing out that our smoothing active set method can deal with the sum-to-one constraints, but the MU method and the Adaptive HT method can not. Our smoothing active set method is flexible to solve the new model in (4.17) with additional sum-to-one constraints. It is interesting to further investigate how to get good estimation of p_j, $j = 1, 2, \ldots, N$, and whether applying our smoothing active set method to this new model can provide even better unmixing results in future.

5. Conclusion remarks. We develop Algorithm 3.1, a novel smoothing active set method, for solving problem (1.1) where the objective function f may be non-Lipschitz continuous. We approximate f by a continuously differentiable function $\tilde{f}$ and employ Algorithm 2.1 for solving the smooth optimization problem (3.5) until the special updating rule holds in the inner loop of Algorithm 3.1. Algorithm 2.1 is a new active set method for linearly constrained smooth optimization, which ensures that for any positive smoothing parameter μ_k, the iterate x^{k+1} satisfies $\|\nabla_\Omega \tilde{f}(x^{k+1}, \mu_k)\| \le \hat{\gamma}\mu_k$. This property is essential for the convergence result of Algorithm 3.1. It is worth noting that convergence results of most existing active set methods for the smooth minimization problem (2.1) are in the sense $\liminf_{k\to\infty} P_\Omega[x^k - \nabla f(x^k)] - x^k = 0$, which does not imply $\liminf_{k\to\infty} \|\nabla_\Omega f(x^k)\| = 0$. See inequality (2.35) and Example 1. Our global convergence result, as well as the nice finite identification property, and the local convergence result makes Algorithm 2.1 not only important for approximately solving subproblems in Algorithm 3.1 for non-Lipschitz minimization problem (1.1), but also advanced for smooth problem (2.1).

Acknowledgements. We are very grateful to Prof. W. W. Hager and the anonymous referees for valuable comments.

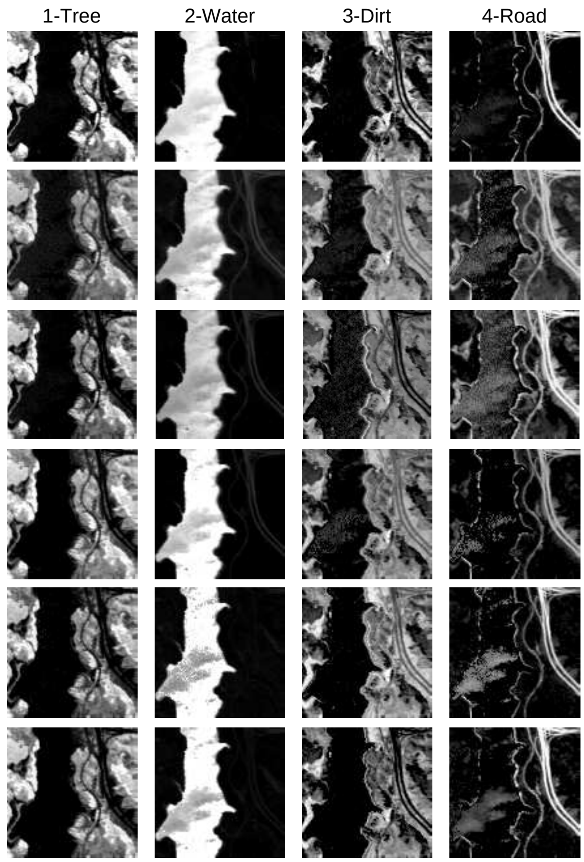

Fig. 2. *Abundance maps from the ground-truth, MU, Adaptive HT, PAO-ASCG-SASCG, PAO-PG-SPG-O, and PAO-PG-SASPG-O (from the first row to the last row sequentially) for four targets in the Jasper Ridge data*.

REFERENCES

[1] N. S. Aybat and G. Iyengar, *A first-order smoothed penalty method for compressed sensing*, SIAM J. Optim., 21 (2011), pp. 287–313.

[2] M. S. Barzaraa, H. D. Sherali and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 2nd ed., John Wiley & Sons, New York, 1993.

[3] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999.

[4] W. Bian and X. Chen, *Linearly constrained non-Lipschitz optimization for image restoration*, SIAM J. Imaging Sci., 8 (2015), pp. 2294–2322.

[5] W. Bian and X. Chen, *Optimality and complexity for constrained optimization problems with nonconvex regularization*, Math. Oper. Res., 42 (2017), pp. 1063–1084.

[6] J. Bolte, A. Danilidis, A. Lewis, and M. Shiota, *Clarke subgradients of stratifiable functins*, SIAM J. Optim., 18 (2007), pp. 556–572.

[7] J. V. Burke, T. Hoheisel, and C. Kanzow, *Gradient consistency for integral-convolution smoothing*, Set-Valued Var. Anal., 21 (2013), pp. 359–376.

[8] P. H. Calamai and J. J. Moré, *Projected gradient method for linearly constrained problems*, Math. Program., 39 (1987), pp. 93–116.

[9] X. Chen, L. Guo, Z. Lu, and J. J. Ye, *An augmented Lagrangian method for non-Lipschitz*

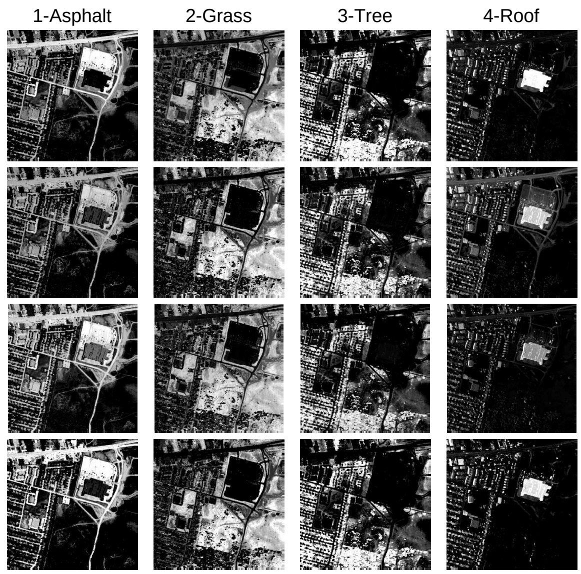

Fig. 3. *Abundance maps from the ground-truth, MU, Adaptive HT and PAO-ASCG-SASCG (from the first row to the last row sequentially) for four targets in the Urban data.*

nonconvex programming, SIAM J. Numer. Anal., 55 (2017), pp. 168–193.
[10] X. Chen, *Smoothing methods for nonsmooth, nonconvex minimization*, Math. Program., 134 (2012), pp. 71–99.
[11] X. Chen, L. Niu, and Y. Yuan, *Optimality conditions and smoothing trust region Newton method for non-Lipschitz optimization*, SIAM J. Optim., 23 (2013), pp. 1528–1552.
[12] X. Chen and W. Zhou, *Smoothing nonlinear conjugate gradient method for image restoration using nonsmooth nonconvex minimization*, SIAM J. Imaging Sci., 3 (2010), pp. 765–790.
[13] Z. Dostál and T. Kozubek, *An optimal algorithm with superrelaxation for minimization of a quadratic functions subject to separable constraints with applications*, Math. Program., 135 (2012), pp. 195-220.
[14] F. Facchinei, A. Fischer, and C. Kanzow, *On the accurate identification of active constraints*, SIAM J. Optim., 9 (1998), pp. 14–32.
[15] B. Fastrich, S. Paterlini, and P. Winker, *Cardinality versus q-norm constraints for index tracking*, Quant. Finance, 14 (2014), pp. 2019–2032.
[16] M. Fukushima, *A modified Frank-Wolfe algorithm for solving the traffic assignment problem*, Transport Res. Part B: Meth., 18 (1984), pp. 169–177.
[17] W. W. Hager and H. Zhang, *A new active set algorithm for box constrained optimization*, SIAM J. Optim., 17 (2006), pp. 526–557.
[18] W. W. Hager and H. Zhang, *An active set algorithm for nonlinear optimization with polyhedral constraints*, Sci. China Math., 59 (2016), pp. 1525–1542.
[19] W. W. Hager and H. Zhang, *An affine scaling method for optimization problems with polyhedral constraints*, Comput. Optim. Appl., 59 (2014), pp. 163-183.
[20] D. Heinz and C.-I. Chang, *Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery*, IEEE Trans. Geosci. Remote Sens., 39 (2001), pp. 529–545.
[21] N. Keskar and A. Wächter, *A limited-memory quasi-Newton algorithm for bound-constrained nonsmooth optimization*, Optim. Method Softw., 34 (2019), pp. 150–171.
[22] J. Kim and H. Park, *Fast nonnegative matrix factorization: An active-set-like method and comparisons*, SIAM J. Sci. Comput., 33 (2011), pp. 3261–3281.
[23] D. Lee and H. Seung, *Algorithms for non-negative matrix factorization*, Adv. Neural Inf. Process. Syst., 13 (2001), pp. 556–562.
[24] C.-J. Lin, *Projected gradient methods for nonnegative matrix factorization*, Neural Comput., 19 (2007), pp. 2756–2779.

[25] C.-J. Lin and J. J. Moré, *Newton's method for large bound-constrained optimization problems*, SIAM J. Optim., 9 (1999), pp. 1100–1127.

[26] Y. Liu, S. Ma, Y. H. Dai, S. Zhang, *A smoothing sequential quadratic programming (SSQP) framework for a class of composite L_q minimization over polyhedron*, Math. Program., 158 (2016), pp. 467–500.

[27] Yu. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program., 103 (2005), pp. 127–152.

[28] M. Nikolova, *Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers*, SIAM J. Numer. Anal., 40 (2002), pp. 965–994.

[29] E. R. Panier, *An active set method for solving linearly constrained nonsmooth optimization problems*, Math. Program., 37 (1987), pp. 269–292.

[30] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, *Hyperspectral unmixing via $L_{1/2}$ sparsity-constrained nonnegative matrix factorization*, IEEE Trans. Geosci. Remote Sens., 49 (2011), pp. 4282–4297.

[31] R. T. Rockafellar and R. J-B. Wets, *Variational Analysis*, Springer, Germany, 1998.

[32] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory*, Society for Industrial and Applied Mathematics, Philadelphia, 2009.

[33] P. Tseng, I. M. Bomze, and W. Schachinger, *A first-order interior-point for linearly constrained smooth optimization*, Math. Program., 127 (2011), pp. 399-424.

[34] W. Wang and M. A. Carreira-Perpinàn, *Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application*, arXiv preprint arXiv: 1309.1541, 2013.

[35] W. Wang and Y. Qian, Adaptive $L_{1/2}$ sparsity-constrained NMF with half-thresholding algorithm for hyperspectral unmixing, IEEE J. Sel Topics Appl. Earth Observ. Remote Sens., 8 (2015), pp. 2618–2631.

[36] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, *A fast algorithm for sparse reconstruction based on shrinkage subspace optimization and continuation*, SIAM J. Comput., 32 (2010), pp. 1832–1857.

[37] Z. Wen, W. Yin, H. Zhang, and D. Goldfarb, *On the convergence of an active set method for ℓ_1 minimization*, Optim. Method Softw., 27 (2012), pp. 1127–1146.

[38] M. Xu, J. J. Ye, and L. Zhang, *Smoothing SQP methods for solving degenerate nonsmooth constrained optimization problems with applications to bilevel programs*, SIAM J. Optim., 25 (2015), pp. 1388–1410.

[39] W. Xu, X. Liu, and Y. Gong, *Document clustering based on nonnegative matrix factorization*, in Proc. 26th Int. Conf. Res. Develop. Inf. Retr. (SIGIR), (2003), pp. 267–273.

[40] Z. Yang, G. Zhou, S. Xie, S. Ding, J. M. Yang, and J. Zhang, *Blind spectral unmixing based on sparse nonnegative matrix factorization*, IEEE Trans. Image Process., 20 (2011), pp. 1112–1125.

[41] C. Zhang and X. Chen, *Smoothing projected gradient method and its application to stochastic linear complementarity problems*, SIAM J. Optim., 20 (2009), pp. 627–649.

[42] C. Zhang, L. Jing, and N. Xiu, *A new active set method for nonnegative matrix factorization*, SIAM J. Sci. Comput., 36 (2014), pp. A2633–A2653.

[43] F. Zhu, Y. Wang, B. Fan, S. Xiang, and G. Meng, *Spectral unmixing via data-guided sparsity*, IEEE Trans. Image Process., 23 (2014), pp. 5412–5427.