

AN EFFICIENT SIEVING-BASED SECANT METHOD FOR SPARSE OPTIMIZATION PROBLEMS WITH LEAST-SQUARES CONSTRAINTS*

QIAN LI[†], DEFENG SUN[†], AND YANCHENG YUAN[†]

Abstract. In this paper, we propose an efficient sieving-based secant method to address the computational challenges of solving sparse optimization problems with least-squares constraints. A level-set method has been introduced in [X. Li, D. F. Sun, and K.-C. Toh, *SIAM J. Optim.*, 28 (2018), pp. 1842–1866] that solves these problems by using the bisection method to find a root of a univariate nonsmooth equation $\varphi(\lambda) = \varrho$ for some $\varrho > 0$, where $\varphi(\cdot)$ is the value function computed by a solution of the corresponding regularized least-squares optimization problem. When the objective function in the constrained problem is a polyhedral gauge function, we prove that (i) for any positive integer k , $\varphi(\cdot)$ is piecewise C^k in an open interval containing the solution λ^* to the equation $\varphi(\lambda) = \varrho$ and that (ii) the Clarke Jacobian of $\varphi(\cdot)$ is always positive. These results allow us to establish the essential ingredients of the fast convergence rates of the secant method. Moreover, an adaptive sieving technique is incorporated into the secant method to effectively reduce the dimension of the level-set subproblems for computing the value of $\varphi(\cdot)$. The high efficiency of the proposed algorithm is demonstrated by extensive numerical results.

Key words. level-set method, secant method, semismooth analysis, adaptive sieving

MSC codes. 90C06, 90C25, 90C90

DOI. 10.1137/23M1594443

1. Introduction. In this paper, we consider the following least-squares constrained optimization problem:

$$(\text{CP}(\varrho)) \quad \min_{x \in \mathbb{R}^n} \{p(x) \mid \|Ax - b\| \leq \varrho\},$$

where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are given data, ϱ is a given parameter satisfying $0 < \varrho < \|b\|$, and $p: \mathbb{R}^n \rightarrow (-\infty, +\infty]$ is a proper closed convex function with $p(0) = 0$ that possesses the property of promoting sparsity. Without loss of generality, we assume that $(\text{CP}(\varrho))$ admits active solutions here.

Let $\lambda > 0$ be a given positive parameter. Compared to the regularized problem of the form

$$(\text{PLS}(\lambda)) \quad \min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} \|Ax - b\|^2 + \lambda p(x) \right\},$$

the constrained optimization problem $(\text{CP}(\varrho))$ is usually preferred in practical modeling since we can regard ϱ as the noise level, which can be estimated in many applications. However, the optimization problem $(\text{CP}(\varrho))$ is perceived to be more challenging to solve in general due to the complicated geometry of the feasible set [1]. Some

*Received by the editors August 15, 2023; accepted for publication (in revised form) March 7, 2024; published electronically June 13, 2024.

<https://doi.org/10.1137/23M1594443>

Funding: The research of the first author was supported in part by Huawei Collaborative Grants “Large Scale Linear Programming Solver” and “Solving Large Scale Linear Programming Models for Production Planning.” The research of the second author was supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, People’s Republic of China (GRF project 15304721 and RGC Senior Research Fellow scheme SRFS2223-5S02). The research of the third author was supported by the Hong Kong Polytechnic University under grant P0038284.

[†]Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Hong Kong (qianxa.li@connect.polyu.hk, defeng.sun@polyu.edu.hk, yancheng.yuan@polyu.edu.hk).

algorithms, such as the alternating direction method of multipliers (ADMM) [13, 11], are applicable to solve $(\text{CP}(\varrho))$. Nevertheless, to obtain an acceptable solution remains challenging for these algorithms. In particular, when applying the ADMM for solving $(\text{CP}(\varrho))$, it is computationally expensive to form the matrix AA^T or to solve the linear systems involved in the subproblems. Recently, dimension reduction techniques, such as adaptive sieving (AS) [38, 37], have achieved some success in solving large-scale sparse optimization problems numerically by exploiting the solution sparsity. But it is still unclear how to apply dimension reduction techniques to $(\text{CP}(\varrho))$ due to the potential infeasibility issue for reduced problems.

A popular approach for solving $(\text{CP}(\varrho))$ and the more general convex constrained optimization problems is the level-set method [33, 34, 1], which has been widely used in many interesting applications [33, 34, 1, 17]. The idea of exchanging the role of the objective function and the constraints, which is the key for the level-set method, has a long history and can date back to Queen Dido's problem (see [26, page 548]). The reader can refer to [1, section 1.3] and the references therein for a discussion of the history of the level-set method. In particular, the level-set method developed in [33, 34] solves the optimization problem $(\text{CP}(\varrho))$ by finding a root of the following univariate nonlinear equation:

$$(E_\phi) \quad \phi(\tau) = \varrho,$$

where $\phi(\cdot)$ is the value function of the following level-set problem:

$$(1.1) \quad \phi(\tau) := \min_{x \in \mathbb{R}^n} \{\|Ax - b\| \mid p(x) \leq \tau\}, \quad \tau \geq 0.$$

Therefore, by executing a root-finding procedure for (E_ϕ) (e.g., the bisection method), one can obtain a solution to $(\text{CP}(\varrho))$ by solving a sequence of problems in the form of (1.1) parameterized by τ . In implementations, one needs an efficient procedure to compute the metric projection of given vectors onto the constraint set $\mathcal{F}_p(\tau) := \{x \in \mathbb{R}^n \mid p(x) \leq \tau, \tau > 0\}$. However, such an efficient computation procedure may not be available. One example can be found in [17], where $p(\cdot)$ is the fused Lasso regularizer [31]. Moreover, it is still not clear to us how to deal with the infeasibility issue when a dimension reduction technique is applied to (1.1).

Recently, Li, Sun, and Toh [17] proposed a level-set method for solving $(\text{CP}(\varrho))$ via solving a sequence of $(\text{P}_{\text{LS}}(\lambda))$. The dual of $(\text{P}_{\text{LS}}(\lambda))$ can be written as

$$(D_{\text{LS}}(\lambda)) \quad \max_{y \in \mathbb{R}^m, u \in \mathbb{R}^n} \left\{ -\frac{1}{2} \|y\|^2 + \langle b, y \rangle - \lambda p^*(u) \mid A^T y - \lambda u = 0 \right\},$$

where $p^*(\cdot)$ is the Fenchel conjugate function of $p(\cdot)$, i.e., $p^*(z) = \sup_{x \in \mathbb{R}^n} \{\langle z, x \rangle - p(x)\}$, $z \in \mathbb{R}^n$. Let $\Omega(\lambda)$ be the solution set to $(\text{P}_{\text{LS}}(\lambda))$. Define the gauge $\Upsilon(\cdot \mid C)$ of a nonempty convex set $C \subseteq \mathbb{R}^n$ as $\Upsilon(x \mid C) := \inf\{\nu \geq 0 \mid x \in \nu C\}$, $x \in \mathbb{R}^n$. Denote $\partial p(0)$ as the subdifferential of $p(\cdot)$ at the origin. In this paper, we assume that

$$(1.2) \quad \lambda_\infty := \Upsilon(A^T b \mid \partial p(0)) \in (0, +\infty)$$

and that for any $\lambda' > 0$, there exists $(y(\lambda'), u(\lambda'), x(\lambda')) \in \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^n$ satisfying the following Karush–Kuhn–Tucker (KKT) system:

$$(KKT) \quad x \in \partial p^*(u), \quad y - b + Ax = 0, \quad A^T y - \lambda' u = 0,$$

where $\partial p^*(\cdot)$ is the subdifferential of $p^*(\cdot)$. Consequently, the solution set $\Omega(\lambda)$ to $(\text{P}_{\text{LS}}(\lambda))$ is nonempty, and $b - Ax(\lambda)$ is invariant for all $x(\lambda) \in \Omega(\lambda)$ since the solution

$(y(\lambda), u(\lambda))$ to $(D_{LS}(\lambda))$ is unique. Based on this fact, Li, Sun, and Toh [17] proposed to solve $(CP(\varrho))$ by finding the root of the following equation:

$$(E_\varphi) \quad \varphi(\lambda) := \|Ax(\lambda) - b\| = \varrho,$$

where $x(\lambda) \in \Omega(\lambda)$ is any solution to $(P_{LS}(\lambda))$. We assume that (E_φ) has at least one solution $\lambda^* > 0$. We then know that any $x(\lambda^*) \in \Omega(\lambda^*)$ is a solution to $(CP(\varrho))$ [17, 10]. There are several advantages to this approach. First, it requires computing the proximal mapping of $p(\cdot)$, which is normally easier than computing the projection over the constraint set of (1.1). Second, efficient algorithms are available to solve the regularized least-squares problem $(P_{LS}(\lambda))$ for a wide class of functions $p(\cdot)$ [16, 17, 18, 39, 2, 13]. More important, this approach is well suited for applying dimension reduction techniques to solve $(P_{LS}(\lambda))$, as can be seen in subsequent sections.

In this paper, we propose an efficient sieving-based secant method for solving $(CP(\varrho))$ by finding the root of (E_φ) . We call our algorithm SMOP, as it is a root-finding-based secant method for solving the optimization problem $(CP(\varrho))$. We focus on the case where $p(\cdot)$ is a gauge function (see [27, section 15]); i.e., $p(\cdot)$ is a nonnegative positively homogeneous convex function with $p(0) = 0$. We start by studying the properties of the value function $\varphi(\cdot)$ and the convergence rates of the secant method for solving (E_φ) . To address the computational challenges for solving $(P_{LS}(\lambda))$ and computing the function value of $\varphi(\cdot)$, we incorporate an AS technique [38, 37] into the secant method to effectively reduce the dimension of $(P_{LS}(\lambda))$. The AS technique can exploit the sparsity of the solution of $(P_{LS}(\lambda))$ so that one can obtain a solution to $(P_{LS}(\lambda))$ by solving a sequence of reduced problems with much smaller dimensions. Extensive numerical results will be presented in this paper to demonstrate the superior performance of the proposed algorithm in solving $(CP(\varrho))$.

The main contributions of this paper can be summarized in the following:

1. When $p(\cdot)$ is a gauge function, we prove that $\varphi(\cdot)$ is (strongly) semismooth for a wide class of instances of $p(\cdot)$ via connecting $D_{LS}(\lambda)$ to a metric projection problem. More important, when $p(\cdot)$ is a polyhedral gauge function, we show that $\varphi(\cdot)$ is locally piecewise C^k on $(0, \lambda_\infty)$ for any integer $k \geq 1$, and for any $\bar{\lambda} \in (0, \lambda_\infty)$, $v > 0$ for any $v \in \partial\varphi(\bar{\lambda})$.
2. Under the assumption that $p(\cdot)$ is a polyhedral gauge function, we show that the secant method converges at least 3-step Q-quadratically for solving (E_φ) , and if $\partial_B\varphi(\lambda^*)$ is a singleton, the secant method converges superlinearly with Q-order at least $(1 + \sqrt{5})/2$. Furthermore, for a general strongly semismooth function $\varphi(\cdot)$, if $\partial\varphi(\lambda^*)$ is a singleton and nondegenerate, the secant method converges superlinearly with R-order of at least $(1 + \sqrt{5})/2$.
3. We propose an efficient sieving-based secant method to address the computational challenges for solving $(CP(\varrho))$. The algorithm incorporates a fast convergent secant method for root finding of (E_φ) , along with an AS technique for effectively reducing the dimension of subproblems in the form of $(P_{LS}(\lambda))$. The efficiency of the proposed algorithm for solving $(CP(\varrho))$ will be demonstrated by extensive numerical results.

The rest of the paper is organized as follows. We will introduce some necessary preliminary results in section 2. We discuss the properties of the value function $\varphi(\cdot)$ in sections 3 and 4. A secant method for solving $(CP(\varrho))$ will be introduced and analyzed in section 5. We will introduce the AS technique in section 6 followed by presenting extensive numerical results in section 7. We conclude the paper in section 8.

Notation. Let $n \geq 1$ be any given integer. Denote the nonnegative orthant and the positive orthant of \mathbb{R}^n as \mathbb{R}_+^n and \mathbb{R}_{++}^n , respectively. We denote $[n] := \{1, 2, \dots, n\}$.

We denote the subvector generated by $x \in \mathbb{R}^n$ indexed by $K \subseteq [n]$ as x_K and the submatrix generated by the columns (rows) of $A \in \mathbb{R}^{m \times n}$ indexed by $K \subseteq [n]$ ($K \subseteq [m]$) as $A_{\cdot K}$ ($A_{K \cdot}$). For any $x \in \mathbb{R}^n$ and any integer $q \geq 1$, the ℓ_q norm of x is defined as $\|x\|_q := \sqrt[q]{|x_1|^q + \dots + |x_n|^q}$. We denote $\|\cdot\| = \|\cdot\|_2$. Let $U \subseteq \mathbb{R}^n$ be an open set. We say that a function $f : U \rightarrow \mathbb{R}$ is C^k for some integer $k \geq 1$ if $f(\cdot)$ is k -times continuously differentiable on U . Let $p : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a proper closed convex function. The proximal mapping of $p(\cdot)$ is defined by

$$\text{Prox}_p(z) := \arg \min_{x \in \mathbb{R}^n} \left\{ p(x) + \frac{1}{2} \|x - z\|^2 \right\}, \quad z \in \mathbb{R}^n.$$

The polar of a gauge function $p(\cdot)$ is defined by

$$p^\circ(y) := \inf \{ \nu \geq 0 \mid \langle y, x \rangle \leq \nu p(x) \quad \forall x \in \mathbb{R}^n \}, \quad y \in \mathbb{R}^n.$$

Let $C \subseteq \mathbb{R}^n$ be a nonempty closed convex set. We use $\Pi_C(\cdot)$ to denote the metric projector over C . The indicator function of C is defined as

$$\delta(x \mid C) := \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{otherwise.} \end{cases}$$

2. Preliminaries. Let \mathcal{X} and \mathcal{Y} be two finite-dimensional real vector spaces each equipped with a scalar product $\langle \cdot, \cdot \rangle$ and its induced norm $\|\cdot\|$. Denote the set of all linear operators from \mathcal{X} to \mathcal{Y} by $\mathcal{L}(\mathcal{X}, \mathcal{Y})$. We first review some preliminary results related to the generalized Jacobians and the semismoothness. Let $\mathcal{O} \subseteq \mathcal{X}$ be an open set and $F : \mathcal{O} \rightarrow \mathcal{Y}$ be a locally Lipschitz continuous function on \mathcal{O} . According to Rademacher’s theorem, $F(\cdot)$ is differentiable (in the sense of Fréchet) almost everywhere on \mathcal{O} . Denote by D_F the set of all points on \mathcal{O} where $F(\cdot)$ is differentiable. Denote $F'(x)$ as the Jacobian of $F(\cdot)$ at $x \in D_F$. Define the B-subdifferential of $F(\cdot)$ at $x \in \mathbb{R}^n$ as

$$(2.1) \quad \partial_B F(x) := \left\{ \lim_{x^k \rightarrow x, x^k \in D_F} F'(x^k) \right\}.$$

The Clarke generalized Jacobian of $F(\cdot)$ at $x \in \mathbb{R}^n$ is then defined as follows [8]:

$$(2.2) \quad \partial F(x) := \text{conv}(\partial_B F(x)).$$

Note that both $\partial_B F(\cdot)$ and $\partial F(\cdot)$ are compact valued and upper-semicontinuous multi-valued functions. For finitely valued convex functions, the Clarke generalized Jacobian coincides with the subdifferential in the sense of convex analysis [8, Proposition 2.2.7]. Now we introduce the concept of G-semismoothness (with respect to a multifunction).

DEFINITION 2.1. Let $\mathcal{O} \subseteq \mathcal{X}$ be an open set, $F : \mathcal{O} \rightarrow \mathcal{Y}$ be a locally Lipschitz continuous function, and $T_F : \mathcal{O} \rightrightarrows \mathcal{L}(\mathcal{X}, \mathcal{Y})$ be a nonempty and compact valued, upper-semicontinuous set-valued mapping. $F(\cdot)$ is said to be G-semismooth at $x \in \mathcal{O}$ with respect to the multifunction T_F if for any $V \in T_F(x + \Delta x)$ with $\Delta x \rightarrow 0$,

$$(2.3) \quad F(x + \Delta x) - F(x) - V \Delta x = o(\|\Delta x\|).$$

Let $\gamma > 0$ be a constant. $F(\cdot)$ is said to be γ -order (strongly if $\gamma = 1$) G-semismooth at $x \in \mathcal{O}$ with respect to T_F if for any $V \in T_F(x + \Delta x)$ with $\Delta x \rightarrow 0$,

$$(2.4) \quad F(x + \Delta x) - F(x) - V \Delta x = O(\|\Delta x\|^{1+\gamma}).$$

$F(\cdot)$ is said to be G -semismooth (resp., γ -order G -semismooth, strongly G -semismooth) on \mathcal{O} with respect to the multifunction T_F if it is G -semismooth (resp., γ -order G -semismooth, strongly G -semismooth) everywhere in \mathcal{O} with respect to the multifunction T_F . All the above definitions of G -semismoothness will be replaced by semismoothness if $F(\cdot)$ happens to be directionally differentiable at the concerned point $x \in \mathcal{O}$.

Let $f: \mathbb{R} \rightarrow \mathbb{R}$ be a real-valued functional. Denote for $x \neq y$ that

$$(2.5) \quad \delta_f(x, y) := (f(x) - f(y)) / (x - y).$$

The following lemma is useful for analyzing the convergence of the secant method. Part (i) of this lemma is from [22, Lemma 2.2], and part (ii) can be proved by following a similar procedure as in the proof of [22, Lemma 2.3].

LEMMA 2.2. Assume that $f: \mathbb{R} \rightarrow \mathbb{R}$ is semismooth at $\bar{x} \in \mathbb{R}$. Denote the lateral derivatives of f at \bar{x} by

$$(2.6) \quad \bar{d}^- := -f'(\bar{x}; -1) \quad \text{and} \quad \bar{d}^+ := f'(\bar{x}; 1).$$

(i) Then the lateral derivatives \bar{d}^- and \bar{d}^+ exist, and

$$\partial_B f(\bar{x}) = \{\bar{d}^-, \bar{d}^+\}.$$

(ii) It holds that

$$(2.7) \quad \bar{d}^- - \delta_f(u, v) = o(1) \quad \text{for all } u \uparrow \bar{x}, v \uparrow \bar{x},$$

$$(2.8) \quad \bar{d}^+ - \delta_f(u, v) = o(1) \quad \text{for all } u \downarrow \bar{x}, v \downarrow \bar{x};$$

moreover, if $f(\cdot)$ is γ -order semismooth at \bar{x} for some $\gamma > 0$, then

$$(2.9) \quad \bar{d}^- - \delta_f(u, v) = O(|u - \bar{x}|^\gamma + |v - \bar{x}|^\gamma) \quad \text{for all } u \uparrow \bar{x}, v \uparrow \bar{x},$$

$$(2.10) \quad \bar{d}^+ - \delta_f(u, v) = O(|u - \bar{x}|^\gamma + |v - \bar{x}|^\gamma) \quad \text{for all } u \downarrow \bar{x}, v \downarrow \bar{x}.$$

3. Properties of the value function $\varphi(\cdot)$. In this section, we first discuss some useful properties of the function $\varphi(\cdot)$. Since $p(\cdot)$ is assumed to be a nonnegative convex function with $p(0) = 0$, we know that $0 \in \partial p(0)$.

PROPOSITION 3.1. Assume that $0 < \lambda_\infty < +\infty$. It holds that

(i) for all $\lambda \geq \lambda_\infty$, $y(\lambda) = b$ and $0 \in \Omega(\lambda)$;

(ii) the value function $\varphi(\cdot)$ is nondecreasing on $(0, +\infty)$, and for any $0 < \lambda_1 < \lambda_2 < +\infty$, $\varphi(\lambda_1) = \varphi(\lambda_2)$ implies that $p(x(\lambda_1)) = p(x(\lambda_2))$, where for any $\lambda > 0$, $x(\lambda)$ is an optimal solution to $(P_{LS}(\lambda))$.

Proof. (i) Since $0 \in \partial p(0)$, for all $\lambda > \lambda_\infty$, it holds that

$$A^T b / \lambda \in \partial p(0),$$

which implies that $0 \in \Omega(\lambda)$. Since $\lambda_\infty > 0$ and $\partial p(0)$ is closed, we know that

$$A^T b / \lambda_\infty \in \partial p(0),$$

which implies that $0 \in \Omega(\lambda_\infty)$. Therefore, for all $\lambda \geq \lambda_\infty$, $0 \in \Omega(\lambda)$ and $y(\lambda) = b$.

(ii) Let $0 < \lambda_1 < \lambda_2 < \infty$ be arbitrarily chosen. Let $x(\lambda_1) \in \Omega(\lambda_1)$ and $x(\lambda_2) \in \Omega(\lambda_2)$. Then we have

$$(3.1) \quad \frac{1}{2} \|Ax(\lambda_1) - b\|^2 + \lambda_1 p(x(\lambda_1)) \leq \frac{1}{2} \|Ax(\lambda_2) - b\|^2 + \lambda_1 p(x(\lambda_2)),$$

$$(3.2) \quad \frac{1}{2} \|Ax(\lambda_2) - b\|^2 + \lambda_2 p(x(\lambda_2)) \leq \frac{1}{2} \|Ax(\lambda_1) - b\|^2 + \lambda_2 p(x(\lambda_1)),$$

which implies that

$$(3.3) \quad (\lambda_1 - \lambda_2)(p(x(\lambda_1)) - p(x(\lambda_2))) \leq 0.$$

Since $\lambda_1 - \lambda_2 < 0$, we know that $p(x(\lambda_1)) \geq p(x(\lambda_2))$. It follows from (3.1) that

$$\frac{1}{2} \|Ax(\lambda_1) - b\|^2 \leq \frac{1}{2} \|Ax(\lambda_2) - b\|^2 + \lambda_1 (p(x(\lambda_2)) - p(x(\lambda_1))) \leq \frac{1}{2} \|Ax(\lambda_2) - b\|^2,$$

which implies that $p(x(\lambda_1)) = p(x(\lambda_2))$ if $\varphi(\lambda_1) = \varphi(\lambda_2)$. This completes the proof of the proposition. \square

Due to Proposition 3.1, we can apply the bisection method to solve (E_φ) , and for any $\epsilon > 0$, we can obtain a solution λ_ϵ satisfying $|\lambda_\epsilon - \lambda^*| \leq \epsilon$ in $O(\log(1/\epsilon))$ iterations, where λ^* is a solution to (E_φ) . In this paper, we will design a more efficient secant method for solving (E_φ) . To achieve this goal, we first study the (strong) semismoothness property of $\varphi(\cdot)$.

In this paper, we focus on the case where $p(\cdot)$ is a gauge function. In most of the applications, $p(\cdot)$ is a norm function, which is automatically a gauge function. We will leave the study of the (strong) semismoothness of $\varphi(\cdot)$ for a general $p(\cdot)$ as future work. Because of its repeated occurrence in this section, we present the following assumption.

Assumption A: Assume that $p(\cdot)$ is a gauge function.

Under Assumption A, $p^*(\cdot) = \delta(\cdot | \partial p(0))$, and the optimization problem $(D_{LS}(\lambda))$ is equivalent to

$$(3.4) \quad \max_{y \in \mathbb{R}^m} \left\{ -\frac{1}{2} \|y\|^2 + \langle b, y \rangle \mid \lambda^{-1} y \in Q \right\},$$

where

$$(3.5) \quad Q := \{z \in \mathbb{R}^m \mid A^T z \in \partial p(0)\}.$$

Then by performing a variable substitution, we have the following useful observation about the solution mapping to (3.4).

PROPOSITION 3.2. *Under Assumption A, for any $0 < \lambda < +\infty$, the unique solution to (3.4) can be written as*

$$(3.6) \quad y(\lambda) = \lambda \Pi_Q(\lambda^{-1} b) = \Pi_{\lambda Q}(b).$$

The following proposition is useful in understanding the semismoothness of $y(\cdot)$ and $\varphi(\cdot)$ even if p is nonpolyhedral. The definition of a tame set and a globally subanalytic set can be found in [3, Definition 2] and [3, Example 2(a)], respectively. Part (ii) of the proposition is generalized from [17, Proposition 1 (iv)] ($p(\cdot)$ is assumed to be a polyhedral gauge function in [17]), and we provide a more explicit proof that does not rely on the piecewise linearity of the solution mapping $y(\cdot)$ in (3.6).

PROPOSITION 3.3. *Under Assumption A, it holds that*

- (i) *the functions $y(\cdot)$ and $\varphi(\cdot)$ are locally Lipschitz continuous on $(0, +\infty)$;*
- (ii) *if $0 < \lambda_\infty < +\infty$, $\varphi(\cdot)$ is strictly increasing on $(0, \lambda_\infty]$;*
- (iii) *if the set Q is tame, $\varphi(\cdot)$ is semismooth on $(0, +\infty)$;*
- (iv) *if Q is globally subanalytic, $\varphi(\cdot)$ is γ -order semismooth on $(0, +\infty)$ for some $\gamma > 0$.*

Proof. For convenience, we denote $\tilde{y}(\lambda) := \Pi_Q(\lambda^{-1}b)$ for any $\lambda > 0$.

(i) Since $\Pi_Q(\cdot)$ is Lipschitz continuous with modulus 1, both $\tilde{y}(\cdot)$ and $y(\cdot)$ are locally Lipschitz continuous on $(0, +\infty)$. Therefore, $\varphi(\cdot) = \|y(\cdot)\|$ is locally Lipschitz continuous on $(0, +\infty)$.

(ii) It follows from Proposition 3.1 that $\varphi(\cdot)$ is nondecreasing. We will now prove that $\varphi(\cdot)$ is strictly increasing on $(0, \lambda_\infty]$. We prove it by contradiction. Assume that there exists $0 < \lambda_1 < \lambda_2 \leq \lambda_\infty$ such that $\varphi(\lambda_1) = \varphi(\lambda_2)$. Let $x(\lambda_1) \in \Omega(\lambda_1)$ and $x(\lambda_2) \in \Omega(\lambda_2)$ be arbitrarily chosen. From Proposition 3.1(ii), we know that

$$p(x(\lambda_1)) = p(x(\lambda_2)),$$

which implies that $x(\lambda_1) \in \Omega(\lambda_2)$ and $x(\lambda_2) \in \Omega(\lambda_1)$. Therefore, we get

$$y(\lambda_1) = b - Ax(\lambda_1) = b - Ax(\lambda_2) = y(\lambda_2).$$

Thus, by using the facts $0 \in \lambda_2 Q$, $y(\lambda_2) = \Pi_{\lambda_2 Q}(b)$, and $\lambda_1^{-1}\lambda_2 y(\lambda_2) = \lambda_1^{-1}\lambda_2 y(\lambda_1) \in \lambda_2 Q$, we obtain from the properties of the metric projector $\Pi_{\lambda_2 Q}(\cdot)$ that

$$\langle b - y(\lambda_2), 0 - y(\lambda_2) \rangle \leq 0, \quad \langle b - y(\lambda_2), \lambda_1^{-1}\lambda_2 y(\lambda_2) - y(\lambda_2) \rangle \leq 0.$$

Therefore,

$$\langle b - y(\lambda_1), y(\lambda_1) \rangle = \langle b - y(\lambda_2), y(\lambda_2) \rangle = 0.$$

Since $\lambda_1 < \lambda_\infty$ and $y(\lambda_1) = \Pi_{\lambda_1 Q}(b)$, we know that $y(\lambda_1) \neq b$. Hence,

$$\langle b - y(\lambda_1), \lambda_\infty^{-1}\lambda_1 b - y(\lambda_1) \rangle = \lambda_\infty^{-1}\lambda_1 \|b - y(\lambda_1)\|^2 > 0.$$

However, $\lambda_\infty^{-1}\lambda_1 b \in \lambda_1 Q$ and $y(\lambda_1) = \Pi_{\lambda_1 Q}(b)$ imply that

$$\langle b - y(\lambda_1), \lambda_\infty^{-1}\lambda_1 b - y(\lambda_1) \rangle \leq 0,$$

which is a contradiction. This contradiction shows that $\varphi(\cdot)$ is strictly increasing on $(0, \lambda_\infty]$.

(iii) Since $\tilde{y}(\cdot)$ is locally Lipschitz continuous on $(0, \lambda_\infty)$, it follows from [3] that $\tilde{y}(\cdot)$ is semismooth on $(0, +\infty)$ if Q is a tame set. Since $\|\cdot\|$ is strongly semismooth, we know that $g_1(\cdot) = \|\tilde{y}(\cdot)\|$ is semismooth on $(0, \lambda_\infty)$. Therefore, $\varphi(\cdot)$ is semismooth on $(0, \lambda_\infty)$ [9, Propositions 7.4.4 and 7.4.8].

(iv) The γ -order semismoothness of $\varphi(\cdot)$ can be proved similarly as for (iii). \square

The above proposition can be used to prove the semismoothness of $\varphi(\cdot)$ for a wide class of functions $p(\cdot)$. For example, the next corollary shows the semismoothness of $\varphi(\cdot)$ when $p(\cdot)$ is the nuclear norm function defined on $\mathbb{R}^{d \times n}$, using the fact that $\partial p(0)$ is linear matrix inequality representable [25].

COROLLARY 3.4. Denote the adjoint of the linear operator $\mathcal{A} : \mathbb{R}^{d \times n} \rightarrow \mathbb{R}^m$ as \mathcal{A}^* . Let $p(\cdot) = \|\cdot\|_*$ be the nuclear norm function defined on $\mathbb{R}^{d \times n}$. Then $Q = \{z \in \mathbb{R}^m \mid \mathcal{A}^*z \in \partial p(0)\}$ is a tame set, and $\Pi_Q(\cdot)$ is semismooth.

Next, we will show that $\varphi(\cdot)$ can be strongly semismooth for a class of important instances of $p(\cdot)$.

PROPOSITION 3.5. Define $\Phi(x) := \frac{1}{2}\|Ax - b\|^2$, $x \in \mathbb{R}^n$, and

$$H(x, \lambda) := x - \text{Prox}_p(x - \lambda^{-1}\nabla\Phi(x)), \quad (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_{++}.$$

For any $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_{++}$, denote $\partial_x H(x, \lambda)$ as the canonical projection of $\partial H(x, \lambda)$ onto \mathbb{R}^n . Under Assumption A, it holds that

- (i) if $\Pi_{\partial p(0)}(\cdot)$ is strongly semismooth and $\partial_x H(\bar{x}, \bar{\lambda})$ is nondegenerate at some $(\bar{x}, \bar{\lambda})$ satisfying $H(\bar{x}, \bar{\lambda}) = 0$, then $y(\cdot)$ and $\varphi(\cdot)$ are strongly semismooth at $\bar{\lambda}$;
- (ii) if $p(\cdot)$ is further assumed to be polyhedral, the function $y(\cdot)$ is piecewise affine, and $\varphi(\cdot)$ is strongly semismooth on \mathbb{R}_{++} .

Proof. (i) It follows from the Moreau identity [27, Theorem 31.5] that for any $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_{++}$,

$$\begin{aligned} H(x, \lambda) &= x - ((x - \lambda^{-1}\nabla\Phi(x)) - \text{Prox}_{p^*}(x - \lambda^{-1}\nabla\Phi(x))) \\ &= \lambda^{-1}\nabla\Phi(x) + \Pi_{\partial p(0)}(x - \lambda^{-1}\nabla\Phi(x)). \end{aligned}$$

The rest of the proof can be obtained from the fact that $\nabla\Phi(\cdot)$ is linear and from the implicit function theorem for semismooth functions [28, 19].

(ii) When $p(\cdot)$ is a polyhedral gauge function, we know that the set Q defined in (3.5) is a convex polyhedral set [27, Theorem 19.3] and that the projector $\Pi_Q(\cdot)$ is piecewise affine [9, Proposition 4.1.4]. Therefore, $y(\cdot)$ is a piecewise affine function on $(0, +\infty)$. Then both $y(\cdot)$ and $\varphi(\cdot)$ are strongly semismooth on $(0, +\infty)$ [9, Propositions 7.4.7, 7.4.4, and 7.4.8]. \square

Remark 3.6. We make some remarks on the assumptions in part (i) of Proposition 3.5. On the one hand, the strong semismoothness of the projector $\Pi_K(\cdot)$ has been proved for some important nonpolyhedral closed convex sets K . In particular, $\Pi_K(\cdot)$ is strongly semismooth if K is the positive semidefinite cone [30], the second-order cone [6], or the ℓ_2 norm ball [39, Lemma 2.1]. On the other hand, the assumption of the nondegeneracy of $\partial_x H(\cdot, \cdot)$ at the concerned point is closely related to the important concept of strong regularity of the KKT system of $(P_{LS}(\lambda))$. One can refer to the monograph [4] and the references therein for a general discussion and to [29, 5] for the semidefinite programming problems.

4. The HS-Jacobian of $\varphi(\cdot)$ for polyhedral gauge functions $p(\cdot)$. In this section, we assume by default that $p(\cdot)$ is a polyhedral gauge function and that $0 < \lambda_\infty < +\infty$. Then the set $\partial p(0)$ is polyhedral [27, Theorem 19.2], which can be assumed without loss of generality to take the form of

$$(4.1) \quad \partial p(0) := \{u \in \mathbb{R}^n \mid Bu \leq d\}$$

for some $B \in \mathbb{R}^{q \times n}$ and $d \in \mathbb{R}^q$.

Inspired by the generalized Jacobian for the projector over a polyhedral set derived by Han and Sun [14], which we call the HS-Jacobian, we will derive the HS-Jacobian of the function $\varphi(\cdot)$. As an important implication, we will prove that the Clarke Jacobian of $\varphi(\cdot)$ at any $\lambda \in (0, \lambda_\infty)$ is positive. Note that the open interval $(0, \lambda_\infty)$ contains the solution λ^* to (E_φ) .

Let $\lambda \in (0, \lambda_\infty)$ be arbitrarily chosen. Let $(y(\lambda), u(\lambda))$ be the unique solution to $(D_{LS}(\lambda))$ with the parameter λ . Here, we denote $(y, u) = (y(\lambda), u(\lambda))$ to simplify our notation and hide the dependency on λ . Then there exists $x \in \Omega(\lambda)$ such that (y, u, x) satisfies the following KKT system:

$$(4.2) \quad u = \Pi_{\partial p(0)}(u + x), \quad y - b + Ax = 0, \quad A^T y - \lambda u = 0.$$

Therefore, u is the unique solution to the optimization problem

$$(4.3) \quad \min_{z \in \mathbb{R}^n} \left\{ \frac{1}{2} \|z - (u + x)\|^2 \mid Bz \leq d \right\},$$

and there exists $\xi \in \mathbb{R}^q$ such that (u, ξ) satisfies the following KKT system for (4.3):

$$(4.4) \quad B^T \xi - x = 0, \quad Bu - d \leq 0, \quad \xi \geq 0, \quad \xi^T (Bu - d) = 0.$$

As a result, there exists $(x, \xi) \in \mathbb{R}^n \times \mathbb{R}^q$ such that (y, u, x, ξ) satisfies the following augmented KKT system:

$$(4.5) \quad \begin{cases} B^T \xi - x = 0, & Bu - d \leq 0, & \xi \geq 0, & \xi^T (Bu - d) = 0, \\ y - b + Ax = 0, & A^T y - \lambda u = 0. \end{cases}$$

Let $M(\lambda)$ be the set of Lagrange multipliers associated with (y, u) defined as

$$M(\lambda) := \{(x, \xi) \in \mathbb{R}^n \times \mathbb{R}^q \mid (y, u, x, \xi) \text{ satisfies (4.5)}\}.$$

Since $x = B^T \xi$, we obtain the following system by eliminating the variable x in (4.5):

$$(4.6) \quad \begin{cases} Bu - d \leq 0, & \xi \geq 0, & \xi^T (Bu - d) = 0, \\ y - b + \widehat{A}\xi = 0, & A^T y - \lambda u = 0. \end{cases}$$

Here, $\widehat{A} = AB^T \in \mathbb{R}^{m \times q}$. Denote

$$(4.7) \quad \widehat{M}(\lambda) := \{\xi \in \mathbb{R}^q \mid (y, u, \xi) \text{ satisfies (4.6)}\}.$$

Then the set $M(\lambda)$ is equivalent to

$$(4.8) \quad M(\lambda) = \{(x, \xi) \in \mathbb{R}^n \times \mathbb{R}^q \mid x = B^T \xi, \xi \in \widehat{M}(\lambda)\}.$$

Denote the active set of u as

$$(4.9) \quad I(u) := \{i \in [q] \mid B_i u - d_i = 0\}.$$

For any $\lambda \in (0, \lambda_\infty)$, we define

$$(4.10) \quad \mathcal{B}(\lambda) := \left\{ K \subseteq [q] \mid \exists \xi \in \widehat{M}(\lambda) \text{ s.t. } \text{supp}(\xi) \subseteq K \subseteq I(u) \text{ and } \text{rank}(\widehat{A}_{:K}) = |K| \right\}.$$

Since the polyhedral set $\widehat{M}(\lambda)$ does not contain a line, this implies that $\widehat{M}(\lambda)$ has at least one extreme point $\bar{\xi}$ [27, Corollary 18.5.3]. Note that $0 < \lambda < \lambda_\infty$ and $x \neq 0$, which implies that $\bar{\xi} \neq 0$ and that $\mathcal{B}(\lambda)$ is nonempty.

Define the HS-Jacobian of $y(\cdot)$ as

$$(4.11) \quad \mathcal{H}(\lambda) := \left\{ h^K \in \mathbb{R}^m \mid h^K = \widehat{A}_{:K} (\widehat{A}_{:K}^T \widehat{A}_{:K})^{-1} d_K, K \in \mathcal{B}(\lambda) \right\}, \quad \lambda \in (0, \lambda_\infty),$$

where d_K is the subvector of d indexed by K . For notational convenience, for any $\lambda \in (0, \lambda_\infty)$ and $K \in \mathcal{B}(\lambda)$, denote

$$(4.12) \quad P^K = I - \widehat{A}_{:K}(\widehat{A}_{:K}^T \widehat{A}_{:K})^{-1} \widehat{A}_{:K}^T.$$

Define

$$(4.13) \quad \mathcal{V}(\lambda) := \{t \in \mathbb{R} \mid t = \lambda \|h\|^2 / \varphi(\lambda), h \in \mathcal{H}(\lambda)\}, \quad \lambda \in \mathcal{D},$$

where $\mathcal{D} = \{\lambda \in (0, \lambda_\infty) \mid \varphi(\lambda) > 0\}$. The following lemma is proved by following the same line as in [14, Lemma 2.1].

LEMMA 4.1. *Let $\bar{\lambda} \in (0, \lambda_\infty)$ be arbitrarily chosen. It holds that*

$$(4.14) \quad y(\bar{\lambda}) = P^K b + \bar{\lambda} h^K \quad \forall h^K \in \mathcal{H}(\bar{\lambda}).$$

Moreover, there exists a positive scalar ς such that $\mathcal{N}(\bar{\lambda}) := (\bar{\lambda} - \varsigma, \bar{\lambda} + \varsigma) \subseteq (0, \lambda_\infty)$, and for all $\lambda \in \mathcal{N}(\bar{\lambda})$,

- (i) $\mathcal{B}(\lambda) \subseteq \mathcal{B}(\bar{\lambda})$ and $\mathcal{H}(\lambda) \subseteq \mathcal{H}(\bar{\lambda})$;
- (ii) $y(\lambda) = y(\bar{\lambda}) + (\lambda - \bar{\lambda})h \quad \forall h \in \mathcal{H}(\lambda)$.

Proof. Choose a sufficiently small $\varsigma > 0$ such that $\mathcal{N}(\bar{\lambda}) := (\bar{\lambda} - \varsigma, \bar{\lambda} + \varsigma) \subseteq (0, \lambda_\infty)$, and let $\lambda \in \mathcal{N}(\bar{\lambda}) \setminus \bar{\lambda}$ be arbitrarily chosen. Denote $(\bar{y}, \bar{u}) = (y(\bar{\lambda}), u(\bar{\lambda}))$ and $(y, u) = (y(\lambda), u(\lambda))$ for notational simplicity.

Let $K \in \mathcal{B}(\bar{\lambda})$ be arbitrarily chosen. From (4.6), there exists $\bar{\xi} \in \widehat{M}(\bar{\lambda})$ with $\text{supp}(\bar{\xi}) \subseteq K \subseteq I(\bar{u})$ such that $(\bar{y}, \bar{u}, \bar{\xi})$ satisfies

$$(4.15) \quad \bar{y} = b - \widehat{A}_{:K} \bar{\xi}_K, \quad B A^T \bar{y} = \widehat{A}^T \bar{y} = \bar{\lambda} B \bar{u}, \quad B_{K^c} \bar{u} = d_K, \quad \xi_{K^c} = 0,$$

where K^c is the complement of K , which implies that

$$\bar{\lambda} d_K = \widehat{A}_{:K}^T (b - \widehat{A}_{:K} \bar{\xi}_K).$$

Since $\widehat{A}_{:K}$ is of full column rank, we have

$$\bar{\xi}_K = (\widehat{A}_{:K}^T \widehat{A}_{:K})^{-1} (\widehat{A}_{:K}^T b - \bar{\lambda} d_K).$$

Consequently, we have

$$\bar{y} = P^K b + \bar{\lambda} h^K,$$

where $h^K = \widehat{A}_{:K} (\widehat{A}_{:K}^T \widehat{A}_{:K})^{-1} d_K \in \mathcal{H}(\bar{\lambda})$.

(i) It follows from Proposition 3.3 that $u(\cdot)$ is locally Lipschitz continuous, which implies that $I(u) \subseteq I(\bar{u})$. Next, we prove that $\mathcal{B}(\lambda) \subseteq \mathcal{B}(\bar{\lambda})$. If not, then there exists a sequence $\{\lambda_k\}_{k \geq 1} \subseteq \mathcal{N}(\bar{\lambda})$ that converges to $\bar{\lambda}$ such that for all k , there is an index set $K^k \in \mathcal{B}(\lambda_k) \setminus \mathcal{B}(\bar{\lambda})$. Denote the solution to (D_{LS}(λ)) with the parameter λ_k as (y^k, u^k) . Since there exist only finitely many choices for the index sets in $\mathcal{B}(\cdot)$, if necessary by taking a subsequence, we assume that the index sets K^k are identical for all $k \geq 1$. Denote the common index set as \tilde{K} . Then the matrix $\widehat{A}_{:\tilde{K}}$ has full column rank, and there exists $\xi^k \in \widehat{M}(\lambda_k)$ (and $(B^T \xi^k, \xi^k) \in M(\lambda_k)$) such that $\text{supp}(\xi^k) \subseteq \tilde{K} \subseteq I(u^k)$, but $\tilde{K} \notin \mathcal{B}(\bar{\lambda})$. Since $I(u^k) \subseteq I(\bar{u})$, there is no $\xi \in \mathcal{B}(\bar{\lambda})$ such that $\text{supp}(\xi) \subseteq \tilde{K}$. However, since $\xi^k \in \widehat{M}(\lambda_k)$, it satisfies

$$(4.16) \quad y^k - b + \widehat{A}_{:\tilde{K}} \xi^k_{\tilde{K}} = 0.$$

As $y(\cdot)$ is locally Lipschitz continuous and $\widehat{A}_{\cdot, \bar{K}}$ is of full column rank, the sequence $\{\xi^k\}_{k \geq 1}$ is bounded. Let $\tilde{\xi}$ be an accumulation point of $\{\xi^k\}_{k \geq 1}$. Then $\tilde{\xi} \in \mathcal{B}(\bar{\lambda})$ and $\text{supp}(\tilde{\xi}) \subseteq \bar{K}$. This is a contradiction. Therefore, $\mathcal{B}(\lambda) \subseteq \mathcal{B}(\bar{\lambda})$. From the definition of $\mathcal{H}(\cdot)$ in (4.11), we also have $\mathcal{H}(\lambda) \subseteq \mathcal{H}(\bar{\lambda})$.

(ii) Let $K \in \mathcal{B}(\lambda)$ be arbitrarily chosen. It follows from (4.14) that

$$y = P^K b + \lambda h^K,$$

where $h^K = \widehat{A}_{\cdot, K}(\widehat{A}_{\cdot, K}^T \widehat{A}_{\cdot, K})^{-1} d_K \in \mathcal{H}(\lambda)$. Since $K \in \mathcal{B}(\lambda) \subseteq \mathcal{B}(\bar{\lambda})$ and $h^K \in \mathcal{H}(\lambda) \subseteq \mathcal{H}(\bar{\lambda})$, in the same vein, we have

$$\bar{y} = P^K b + \bar{\lambda} h^K.$$

As a result, for all $h \in \mathcal{H}(\lambda)$,

$$y = \bar{y} + (\lambda - \bar{\lambda})h.$$

We complete the proof of the lemma. \square

Next, we prove the nondegeneracy of $\partial\varphi(\bar{\lambda})$ for any $\bar{\lambda} \in (0, \lambda_\infty)$, which is important for analyzing the convergence rates of the secant method for solving (E_φ) .

THEOREM 4.2. *Let $p(\cdot)$ be a polyhedral gauge function. Assume that $0 < \lambda_\infty < +\infty$. For any $\bar{\lambda} \in (0, \lambda_\infty)$, it holds that*

- (i) *for any integer $k \geq 1$, the function $\varphi(\cdot)$ is piecewise C^k in an open interval containing $\bar{\lambda}$;*
- (ii) *all $v \in \partial\varphi(\bar{\lambda})$ are positive.*

Proof. Choose a sufficiently small $\varsigma > 0$ such that $\mathcal{N}(\bar{\lambda}) = (\bar{\lambda} - \varsigma, \bar{\lambda} + \varsigma) \subseteq (0, \lambda_\infty)$ and $\mathcal{B}(\lambda) \subseteq \mathcal{B}(\bar{\lambda})$ for any $\lambda \in \mathcal{N}(\bar{\lambda})$. Let $\lambda \in \mathcal{N}(\bar{\lambda}) \setminus \bar{\lambda}$ and $K \in \mathcal{B}(\lambda)$ be arbitrarily chosen. Denote

$$h^K = \widehat{A}_{\cdot, K}(\widehat{A}_{\cdot, K}^T \widehat{A}_{\cdot, K})^{-1} d_K.$$

Then we have $h^K \in \mathcal{H}(\lambda) \subseteq \mathcal{H}(\bar{\lambda})$.

Now we prove part (i) of the theorem. From the fact that

$$\langle P^K b, h^K \rangle = 0$$

and Lemma 4.1, we know that

$$\varphi(\lambda) = \sqrt{\|P^K b\|^2 + \lambda^2 \|h^K\|^2} \quad \text{and} \quad \varphi(\bar{\lambda}) = \sqrt{\|P^K b\|^2 + (\bar{\lambda})^2 \|h^K\|^2}.$$

Define $\varphi^K : \mathbb{R} \rightarrow \mathbb{R}_+$ by

$$(4.17) \quad \varphi^K(s) := \sqrt{\|P^K b\|^2 + s^2 \|h^K\|^2}, \quad s \in \mathbb{R}.$$

From Proposition 3.3, we know that $\varphi(\cdot)$ is strictly increasing on $(0, \lambda_\infty]$. Therefore, it holds that

$$(4.18) \quad \varphi^K(\lambda) = \varphi(\lambda) \neq \varphi(\bar{\lambda}) = \varphi^K(\bar{\lambda}),$$

which implies that $h^K \neq 0$. Since $K \in \mathcal{B}(\lambda)$ is arbitrarily chosen, we obtain that

$$(4.19) \quad h^K \neq 0 \quad \forall K \in \mathcal{B}(\lambda).$$

Thus, for any integer $k \geq 1$, $\varphi^K(\cdot)$ is C^k on $\mathcal{N}(\bar{\lambda})$.

Denote $\bar{\mathcal{B}} = \bigcup_{\lambda \in \mathcal{N}(\bar{\lambda}) \setminus \bar{\lambda}} \mathcal{B}(\lambda)$. We know that $\bar{\mathcal{B}} \subseteq \mathcal{B}(\bar{\lambda})$ and that $|\bar{\mathcal{B}}|$ is finite. Moreover,

$$(4.20) \quad \varphi(s) \in \{\varphi^K(s)\}_{K \in \bar{\mathcal{B}}} \quad \forall s \in \mathcal{N}(\bar{\lambda}),$$

which implies that for any $k \geq 1$, $\varphi(\cdot)$ is piecewise C^k on $\mathcal{N}(\bar{\lambda})$.

Next, we prove part (ii) of the theorem. It follows from (4.19) that for any $K \in \bar{\mathcal{B}}$ and $\lambda > 0$,

$$(4.21) \quad h^K \neq 0 \quad \text{and} \quad (\varphi^K)'(\lambda) = \lambda \|h^K\|^2 / \varphi^K(\lambda) > 0.$$

By [8, Theorem 2.5.1] and the upper semicontinuity of $\partial_B \varphi(\cdot)$, we have

$$(4.22) \quad \partial \varphi(\bar{\lambda}) \subseteq \text{conv}(\{\bar{\lambda} \|h^K\|^2 / \varphi(\bar{\lambda}) \mid K \in \bar{\mathcal{B}}\}),$$

which implies that

$$v > 0 \quad \forall v \in \partial \varphi(\bar{\lambda}).$$

We complete the proof of the theorem. □

5. A secant method for $(\text{CP}(\rho))$. In this section, we will design a fast convergent secant method for solving (E_φ) and prove its convergence rates.

5.1. A fast convergent secant method for semismooth equations. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be a locally Lipschitz continuous function which is semismooth at a solution x^* to the following equation:

$$(5.1) \quad f(x) = 0.$$

In this section, we analyze the convergence of the secant method described in Algorithm 5.1 with two generic starting points x^{-1} and x^0 .

The convergence results of Algorithm 5.1 are given in the following proposition. The proof can be obtained by following the procedure in the proof of [22, Theorem 3.2].

PROPOSITION 5.1. *Suppose that $f : \mathbb{R} \rightarrow \mathbb{R}$ is semismooth at a solution x^* to (5.1). Let d^- and d^+ be the lateral derivatives of f at x^* as defined in (2.6). If d^- and d^+ are both positive (or negative), then there are two neighborhoods \mathcal{U} and \mathcal{N} of x^* , $\mathcal{U} \subseteq \mathcal{N}$, such that for $x^{-1}, x^0 \in \mathcal{U}$, Algorithm 5.1 is well defined and produces a sequence of iterates $\{x^k\}$ such that $\{x^k\} \subseteq \mathcal{N}$. The sequence $\{x^k\}$ converges to x^* 3-step Q -superlinearly, i.e., $|x^{k+3} - x^*| = o(|x^k - x^*|)$. Moreover, it holds that*

Algorithm 5.1 A secant method for solving (5.1).

- 1: **Input:** $x^{-1}, x^0 \in \mathbb{R}$.
- 2: **Initialization:** Set $k = 0$.
- 3: **while** $f(x^k) \neq 0$ **do**
- 4: **Step 1.** Compute

$$(5.2) \quad x^{k+1} = x^k - (\delta_f(x^k, x^{k-1}))^{-1} f(x^k).$$

- 5: **Step 2.** Set $k = k + 1$.
 - 6: **end while**
 - 7: **Output:** x^k .
-

TABLE 1

The numerical performance of finding the zero of (5.3). Case I: $\beta = 1.1$, $d^+ = 1.1$, $d^- = 1$, and $\alpha = 0.1$; Case II: $\beta = 1.5$, $d^+ = 1.5$, $d^- = 1$, and $\alpha = 0.5$; Case III: $\beta = 2.1$, $d^+ = 2.1$, $d^- = 1$, and $\alpha = 1.1$.

Case	Iter	1	2	3	4	5	6	7	8
I	x	-5.1e-5	-4.3e-6	2.2e-10	-2.2e-11	-1.8e-12	4.1e-23	-4.1e-24	-3.4e-25
II	x	-5.1e-5	-1.7e-5	8.4e-10	-4.2e-10	-1.1e-10	4.5e-20	-2.2e-20	-5.6e-21
III	x	-5.1e-5	-2.6e-5	1.3e-9	-1.5e-9	-5.1e-10	7.4e-19	-8.2e-19	-2.8e-19

- (i) $|x^{k+1} - x^*| \leq \frac{|d^+ - d^- + o(1)|}{\min\{|d^+|, |d^-|\} + o(1)} |x^k - x^*|$ for $k \geq 0$;
- (ii) if $\alpha := \frac{|d^+ - d^-|}{\min\{|d^+|, |d^-|\}} < 1$, then $\{x^k\}$ converges to x^* Q-linearly with Q-factor α ;
- (iii) if f is γ -order semismooth at x^* for some $\gamma > 0$, then $|x^{k+3} - x^*| = O(|x^k - x^*|^{1+\gamma})$ for sufficiently large k ; the sequence $\{x^k\}$ converges to x^* 3-step quadratically if f is strongly semismooth at x^* .

Here, we only consider the case for $d^+ \cdot d^- > 0$ since the function $\varphi(\cdot)$ we are interested in is nondecreasing. For the case $d^+ \cdot d^- < 0$, one can refer to [22, Theorem 3.3].

When $|d^+ - d^-|$ is small and f is strongly semismooth, we know from Proposition 5.1 that the secant method converges with a fast linear rate and 3-step Q-quadratic rate. We provide a numerical example slightly modified from [22, equation (3.15)] to illustrate the convergence rates obtained in Proposition 5.1. We test Algorithm 5.1 with $x^{-1} = 0.01$ and $x^0 = 0.005$ for finding the zero $x^* = 0$ of

$$(5.3) \quad f(x) = \begin{cases} x(x+1) & \text{if } x < 0, \\ -\beta x(x-1) & \text{if } x \geq 0, \end{cases}$$

where β is chosen from $\{1.1, 1.5, 2.1\}$. The numerical results are shown in Table 1 and coincide with our theoretical results.

Note that [22, Lemma 4.1] implies that the sequence $\{x^k\}$ generated by Algorithm 5.1 converges suplinearly with R-order at least $\sqrt[3]{2}$. Next, we will prove that the sequence $\{x^k\}$ generated by Algorithm 5.1 converges superlinearly to a solution x^* to (5.1) with R-order at least $(1 + \sqrt{5})/2$ when f is strongly semismooth at x^* and $\partial f(x^*)$ is a singleton and nondegenerate.

PROPOSITION 5.2. *Let x^* be a solution to (5.1). Let $\{x^k\}$ be the sequence generated by Algorithm 5.1 for solving (5.1). For $k \geq -1$, denote $e_k := x^k - x^*$, and assume that $|e_k| > 0$. For $k \geq -1$, denote $c_k := |e_k|/(|e_{k-1}||e_{k-2}|)$. Assume that $\partial f(x^*)$ is a singleton and nondegenerate. It holds that*

- (i) if f is semismooth at x^* , the sequence $\{x^k\}$ converges to x^* Q-superlinearly;
- (ii) if f is strongly semismooth at x^* , then either one of the following two properties is satisfied: (1) $\{x^k\}$ converges to x^* superlinearly with Q-order at least $(1 + \sqrt{5})/2$; (2) $\{x^k\}$ converges to x^* superlinearly with R-order at least $(1 + \sqrt{5})/2$, and for any constant $C > 0$, there exists a subsequence $\{c_{i_k}\}$ satisfying $c_{i_k} < C i_k^{-i_k}$.

Proof. Let \mathcal{N} and \mathcal{U} be the neighborhoods of x^* specified in Proposition 5.1. Assume that $x^{-1}, x^0 \in \mathcal{U}$. Then Algorithm 5.1 is well defined, and it generates a

sequence $\{x^k\} \subseteq \mathcal{N}$ which converges to x^* . Denote $\partial f(x^*) = \{v^*\}$ for some $v^* \neq 0$. Let d^- and d^+ be the lateral derivatives of f at x^* as defined in (2.6). Then

$$d^+ = d^- = v^*.$$

Let K_1 be a sufficiently large integer. For all $k \geq K_1$, we have

$$(5.4) \quad e_{k+1} = \delta_f(x^k, x^{k-1})^{-1} [\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)] e_k.$$

(i) Assume that f is semismooth at x^* . We estimate

$$\delta_f(x^k, x^{k-1})^{-1} [\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)]$$

by considering the following two cases:

(i-a) $x^k, x^{k-1} > x^*$ or $x^k, x^{k-1} < x^*$. From Lemma 2.2, we obtain that

$$\begin{aligned} |e_{k+1}| &= |\delta_f(x^k, x^{k-1})^{-1} [\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)] e_k| \\ &= |(v^* + o(1))^{-1} [(v^* + o(1)) - (v^* + o(1))] e_k| \\ &= o(|e_k|). \end{aligned}$$

(i-b) $x^{k-1} < x^* < x^k$ or $x^k < x^* < x^{k-1}$. We will consider the first case. The second case can be treated similarly. By Lemma 2.2, it holds that

$$\begin{aligned} \delta_f(x^k, x^{k-1}) &= \frac{f(x^k) - f(x^*) + f(x^*) - f(x^{k-1})}{x^k - x^{k-1}} \\ &= \frac{(v^* e_k + o(|e_k|)) - (v^* e_{k-1} + o(|e_{k-1}|))}{x^k - x^{k-1}} \\ &= v^* + o(1). \end{aligned}$$

Therefore,

$$\begin{aligned} |e_{k+1}| &= |\delta_f(x^k, x^{k-1})^{-1} [\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)] e_k| \\ &= |(v^* + o(1))^{-1} [(v^* + o(1)) - (v^* + o(1))] e_k| \\ &= o(|e_k|). \end{aligned}$$

Thus, we prove that the sequence $\{x^k\}$ converges to x^* Q-superlinearly.

(ii) Now assume that f is strongly semismooth at x^* . We build the recursion for e_k for sufficiently large integers k by considering the following two cases:

(ii-a) $x^k, x^{k-1} > x^*$ or $x^k, x^{k-1} < x^*$. From Lemma 2.2, we obtain

$$\begin{aligned} |e_{k+1}| &= |\delta_f(x^k, x^{k-1})^{-1} [\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)] e_k| \\ &= |v^* + O(|e_k| + |e_{k-1}|)^{-1} |O(|e_k| + |e_{k-1}|)| |e_k| \\ &= O(|e_k| (|e_k| + |e_{k-1}|)). \end{aligned}$$

(ii-b) $x^{k-1} < x^* < x^k$ or $x^k < x^* < x^{k-1}$. We will consider the first case. The second case can be treated similarly. By Lemma 2.2, it holds that

$$\begin{aligned} \delta_f(x^k, x^{k-1}) &= \frac{f(x^k) - f(x^*) + f(x^*) - f(x^{k-1})}{x^k - x^{k-1}} \\ &= \frac{(v^* e_k + O(|e_k|^2)) - (v^* e_{k-1} + O(|e_{k-1}|^2))}{x^k - x^{k-1}} \\ &= v^* + O(|e_k| + |e_{k-1}|) \end{aligned}$$

and

$$\begin{aligned} |e_{k+1}| &= |\delta_f(x^k, x^{k-1})^{-1}[\delta_f(x^k, x^{k-1}) - \delta_f(x^k, x^*)]e_k| \\ &= |v^* + O(|e_k| + |e_{k-1}|)|^{-1}|O(|e_k| + |e_{k-1}|)|e_k| \\ &= O(|e_k|(|e_k| + |e_{k-1}|)). \end{aligned}$$

Therefore, for sufficiently large integers k , we have

$$(5.5) \quad |e_{k+1}| = O(|e_k|^2 + |e_k||e_{k-1}|)$$

and

$$(5.6) \quad \limsup_{k \rightarrow \infty} c_k = O(1/|v^*|) < +\infty.$$

Then there exists a constant $\bar{C} > 0$ and a positive integer K_2 such that

$$|e_{k+1}| \leq \bar{C}|e_k||e_{k-1}| \quad \forall k \geq K_2.$$

Therefore, it follows from [20, Theorem 9.2.9] that $\{x^k\}$ converges to x^* with R-order at least $(1 + \sqrt{5})/2$.

If there exists a constant $\underline{C} > 0$ such that $c_k \geq \underline{C}k^{-k}$ for all k sufficiently large, it follows from [21, Corollary 3.1] that $\{x^k\}$ converges to x^* Q-superlinearly with Q-order at least $(1 + \sqrt{5})/2$. We complete the proof. \square

PROPOSITION 5.3. *Let $p(\cdot)$ be a polyhedral gauge function and λ^* be the solution to (E_φ) . Assume that $0 < \lambda_\infty < +\infty$. If $\partial\varphi(\lambda^*)$ is a singleton, the sequence $\{\lambda_k\}$ generated by Algorithm 5.1 for solving (E_φ) converges to λ^* Q-superlinearly with Q-order at least $(1 + \sqrt{5})/2$.*

Proof. The assumption that $\partial\varphi(\lambda^*)$ is a singleton implies that $\varphi(\cdot)$ is strictly differentiable at λ^* [8, Proposition 2.2.4]. It follows from Theorem 4.2 that $\varphi'(\lambda^*) > 0$ and that $\varphi(\cdot)$ is piecewise C^k for any positive integer $k \geq 1$ in a neighborhood of λ^* .

Choose a sufficiently small $\varsigma > 0$ such that $\mathcal{N}(\lambda^*) := (\lambda^* - \varsigma, \lambda^* + \varsigma) \subseteq (0, \lambda_\infty)$ and $\mathcal{B}(\lambda) \subseteq \mathcal{B}(\lambda^*)$ for any $\lambda \in \mathcal{N}(\lambda^*)$. Denote $\bar{\mathcal{B}} = \bigcup_{\lambda \in \mathcal{N}(\lambda^*) \setminus \lambda^*} \mathcal{B}(\lambda)$. Let $K \in \bar{\mathcal{B}}$ be arbitrarily chosen. Define $\varphi^K : \mathbb{R} \rightarrow \mathbb{R}_+$ by

$$\varphi^K(s) := \sqrt{\|P^K b\|^2 + s^2 \|h^K\|^2}, \quad s \in \mathbb{R}.$$

By choosing a smaller ς if necessary, we assume that $\{\varphi^K\}_{K \in \bar{\mathcal{B}}}$ is a minimal local representation for $\varphi(\cdot)$ at λ^* . Therefore, it follows from [24, Theorem 2] that

$$\varphi'(\lambda^*) = (\varphi^K)'(\lambda^*) = \lambda^* \|h^K\|^2 / \varphi(\lambda^*) \quad \forall K \in \bar{\mathcal{B}}.$$

Since $\varphi'(\lambda^*) > 0$, we have

$$\|h^K\| = \|h^{K'}\|, \quad \|P^K b\| = \|P^{K'} b\| \quad \forall K, K' \in \bar{\mathcal{B}}.$$

Therefore, $\varphi(\cdot)$ is C^k on $\mathcal{N}(\lambda^*)$ for any integer $k \geq 1$. It follows from [32, Example 6.1] that $\{\lambda_k\}$ converges to λ^* Q-superlinearly with Q-order at least $(1 + \sqrt{5})/2$. \square

We give the following example to show that a function satisfying the assumptions in (ii) of Proposition 5.2 are not necessarily piecewise smooth:

$$(5.7) \quad f(x) = \begin{cases} \kappa x & \text{if } x < 0, \\ -\frac{1}{3} \left(\frac{1}{4^k}\right) + \left(1 + \frac{1}{2^k}\right)x & \text{if } x \in \left[\frac{1}{2^{k+1}}, \frac{1}{2^k}\right] \\ 2x - \frac{1}{3} & \text{if } x > 1, \end{cases} \quad \forall k \geq 0,$$

where κ is a given constant.

PROPOSITION 5.4. *The function f defined in (5.7) is strongly semismooth at $x = 0$ but not piecewise smooth in the neighborhood of $x = 0$.*

Proof. By the construction of $f(\cdot)$, we know that it is not piecewise smooth in the neighborhood of $x = 0$ since there are infinitely many nondifferentiable points. Next, we show that f is strongly semismooth at $x = 0$.

First, it is not difficult to verify that f is Lipschitz continuous with modulus $L = \max\{|\kappa|, 2\}$. Second, we know that $f'(0; -1) = \kappa$, and for any integer $k \geq 0$, it holds that

$$1 + \frac{1}{3 \times 2^k} \leq f(x)/x \leq \left(1 + \frac{1}{3 \times 2^{k-1}}\right) \quad \forall x \in \left[\frac{1}{2^{k+1}}, \frac{1}{2^k}\right],$$

which implies that

$$f'(x; 1) = \lim_{x \downarrow 0} f(x)/x = 1.$$

Therefore, f is directionally differentiable at $x = 0$. Note that both $\partial f(0)$ and $\partial_{\mathbb{B}} f(0)$ are singleton when $\kappa = 1$.

Next, we show that $f(\cdot)$ is strongly G-semismooth at $x = 0$. On the one hand, for any $x < 0$, we have

$$|f(x) - f(0) - \kappa x| = |\kappa x - \kappa x| = 0.$$

On the other hand, for any integer $k \geq 1$ and $x \in [\frac{1}{2^{k+1}}, \frac{1}{2^k}]$, we know that

$$1 + 2^{-k} \leq |v| \leq 1 + 2^{1-k} \quad \forall v \in \partial f(x),$$

which implies that

$$|f(x) - f(0) - v(x)x| = \left| -\frac{1}{3} \left(\frac{1}{4^k}\right) + \left(1 + \frac{1}{2^k}\right)x - v(x)x \right| \leq \frac{1}{2^{k-1}}x \leq 4x^2.$$

Therefore,

$$|f(x) - f(0) - v(x)x| = O(|x|^2) \quad \forall x \rightarrow 0.$$

The proof of the proposition is completed. \square

We end this subsection by illustrating the numerical performance of Algorithm 5.1 for finding the root of $f(x)$ given in (5.7) with $\kappa = 1$. Note that $x^* = 0$ is the unique solution. In Algorithm 5.1, we choose $x^0 = 0.5$ and $x^{-1} = x^0 + 0.1 \times f(0.5)^2 = 0.545$. The numerical results are shown in Table 2.

We can observe that the generated sequence $\{x_k\}$ converges to the solution $x^* = 0$ superlinearly with R-order at least $(1 + \sqrt{5})/2$.

TABLE 2
The numerical performance of Algorithm 5.1 on finding the zero of (5.7).

Iter	1	2	3	4	5	6	7	8
x	1.7e-1	3.6e-2	4.0e-3	1.0e-4	2.7e-7	2.0e-11	4.0e-18	6.1e-29
$f(x)$	1.9e-1	3.7e-2	4.0e-3	1.0e-4	2.7e-7	2.0e-11	4.0e-18	6.1e-29

Algorithm 5.2 A globally convergent secant method for $(\text{CP}(\varrho))$.

- 1: **Input:** $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^n$, $\mu \in (0, 1)$, $\lambda_{-1}, \lambda_0, \lambda_1$ in $(0, \lambda_\infty)$ satisfying $\varphi(\lambda_0) > \varrho$, and $\varphi(\lambda_{-1}) < \varrho$.
- 2: **Initialization:** Set $i = 0$, $\underline{\lambda} = \lambda_{-1}$, and $\bar{\lambda} = \lambda_0$.
- 3: **for** $k = 1, 2, \dots$ **do**
- 4: Compute

$$(5.8) \quad \hat{\lambda}_{k+1} = \lambda_k - \frac{\lambda_k - \lambda_{k-1}}{\varphi(\lambda_k) - \varphi(\lambda_{k-1})} (\varphi(\lambda_k) - \varrho).$$

- 5: **if** $\hat{\lambda}_{k+1} \in [\lambda_{-1}, \lambda_0]$ **then**
 - 6: Compute $x(\hat{\lambda}_{k+1})$ and $\varphi(\hat{\lambda}_{k+1})$. Set $i = i + 1$.
 - 7: **if** either (i) or (ii) holds: (i) $i \geq 3$ and $|\varphi(\hat{\lambda}_{k+1}) - \varrho| \leq \mu |\varphi(\lambda_{k-2}) - \varrho|$ (ii) $i < 3$, **then** set $\lambda_{k+1} = \hat{\lambda}_{k+1}$, $x(\lambda_{k+1}) = x(\hat{\lambda}_{k+1})$; **else** go to line 9.
 - 8: **else**
 - 9: **if** $\varphi(\hat{\lambda}_{k+1}) > \varrho$, **then** set $\bar{\lambda} = \min\{\bar{\lambda}, \hat{\lambda}_{k+1}\}$; **else** set $\underline{\lambda} = \max\{\underline{\lambda}, \hat{\lambda}_{k+1}\}$.
 - 10: Set $\lambda_{k+1} = 1/2(\bar{\lambda} + \underline{\lambda})$. Compute $x(\lambda_{k+1})$ and $\varphi(\lambda_{k+1})$. Set $i = 0$.
 - 11: **end if**
 - 12: **if** $\varphi(\lambda_{k+1}) > \varrho$, **then** set $\bar{\lambda} = \min\{\bar{\lambda}, \lambda_{k+1}\}$; **else** set $\underline{\lambda} = \max\{\underline{\lambda}, \lambda_{k+1}\}$.
 - 13: **end for**
 - 14: **Output:** $x(\lambda_k)$ and λ_k .
-

5.2. A globally convergent secant method for $(\text{CP}(\varrho))$. In this section, we propose a globally convergent secant method for solving $(\text{CP}(\varrho))$ via finding the root of (E_φ) . The algorithm is described in Algorithm 5.2.

THEOREM 5.5. *Let $p(\cdot)$ be a gauge function, and assume that $0 < \lambda_\infty < +\infty$. Denote λ^* as the solution to (E_φ) . Then Algorithm 5.2 is well defined, and the sequences $\{\lambda_k\}$ and $\{x(\lambda_k)\}$ converge to λ^* and a solution $x(\lambda^*)$ to $(\text{CP}(\varrho))$, respectively. Denote $e_k = \lambda_k - \lambda^*$ for all $k \geq 1$. Suppose that both d^+ and d^- of $\varphi(\cdot)$ at λ^* as defined in (2.6) are positive. The following properties hold for all sufficiently large integer k :*

- (i) *If $\varphi(\cdot)$ is semismooth at λ^* , then $|e_{k+3}| = o(|e_k|)$.*
- (ii) *If $\varphi(\cdot)$ is γ -order semismooth at λ^* for some $\gamma > 0$, then $|e_{k+3}| = O(|e_k|^{1+\gamma})$.*
- (iii) *If $\partial\varphi(\lambda^*)$ is a singleton and $\varphi(\cdot)$ is semismooth at λ^* , then $\{e_k\}$ converges to zero Q -superlinearly; if $p(\cdot)$ is further assumed to be polyhedral and $\partial\varphi(\lambda^*)$ is a singleton, then $\{e_k\}$ converges to zero superlinearly with Q -order $(1+\sqrt{5})/2$.*

Proof. When $p(\cdot)$ is a gauge function, we know from Proposition 3.3 that $\varphi(\cdot)$ is strictly increasing on $(0, \lambda_\infty)$, which implies that the sequences $\{\hat{\lambda}_k\}$ and $\{\lambda_k\}$ generated in Algorithm 5.2 are well defined. For any $k \geq 1$, if we run the algorithm for three more iterations, then it holds that either (a) $(\bar{\lambda} - \underline{\lambda})$ will reduce at least half or (b) $|\varphi(\lambda_{k+3}) - \varrho| \leq \mu |\varphi(\lambda_k) - \varrho|$. Therefore, the sequence $\{\lambda_k\}$ will converge to λ^* . Suppose that $\varphi(\cdot)$ is semismooth at λ^* and that both d^+ and d^- are positive. We know from Proposition 5.1 that there exists a positive integer k_{\max} such that for all $k \geq k_{\max}$, $\hat{\lambda}_k \in [\lambda_{-1}, \lambda_0]$ and

$$(5.9) \quad |\hat{\lambda}_{k+3} - \lambda^*| = o(|\lambda_k - \lambda^*|).$$

Therefore, it follows from Lemma 2.2 that

$$\frac{|\varphi(\hat{\lambda}_{k+3}) - \varrho|}{|\varphi(\lambda_k) - \varrho|} = \frac{\delta_\varphi(\hat{\lambda}_{k+3}, \lambda^*)}{\delta_\varphi(\lambda_k, \lambda^*)} \times \frac{|\hat{\lambda}_{k+3} - \lambda^*|}{|\lambda_k - \lambda^*|} \leq \frac{\max\{d^+, d^-\} + o(1)}{\min\{d^+, d^-\} + o(1)} \times o(1).$$

Thus, for all $k \geq k_{\max}$,

$$|\varphi(\hat{\lambda}_{k+3}) - \varrho| \leq \mu |\varphi(\lambda_k) - \varrho|.$$

The rest of the proof of this theorem follows from Propositions 5.1 and 5.3. \square

To better illustrate the efficiency of Algorithm 5.2, we will compare its performance to the HS-Jacobian-based semismooth Newton method for solving (E_φ) on the least-squares constrained Lasso problem, where the HS-Jacobian is available. The following proposition shows that for the least-squares constrained Lasso problem, $\partial_{\text{HS}}\varphi(\bar{\lambda})$ is positive for any $\bar{\lambda} \in (0, \lambda_\infty)$.

PROPOSITION 5.6. *Suppose that $p(\cdot)$ is a polyhedral gauge function and that $\partial p(0)$ has the expression as in (4.1). Assume that $0 < \lambda_\infty < +\infty$, and let $\bar{\lambda} \in (0, \lambda_\infty)$ be arbitrarily chosen. Let $\mathcal{B}(\bar{\lambda})$ and $\mathcal{V}(\bar{\lambda})$ be the sets defined as in (4.10) and (4.13) for $\lambda = \bar{\lambda}$. If $d_K \neq 0$ for all $K \in \mathcal{B}(\bar{\lambda})$, then $v > 0$ for all $v \in \mathcal{V}(\bar{\lambda})$. Moreover, $d_K \neq 0$ for all $K \in \mathcal{B}(\bar{\lambda})$ when $p(\cdot) = \|\cdot\|_1$.*

Proof. Recall that $\mathcal{B}(\bar{\lambda})$ is nonempty. Let $K \in \mathcal{B}(\bar{\lambda})$ be arbitrarily chosen. We know that $\hat{A}_{:K}$ is of full column rank. Denote $h^K = \hat{A}_{:K}(\hat{A}_{:K}^T \hat{A}_{:K})^{-1} d_K \in \mathcal{H}(\bar{\lambda})$. Since $d_K \neq 0$, it holds that

$$\|h^K\|^2 = \langle \hat{A}_{:K}(\hat{A}_{:K}^T \hat{A}_{:K})^{-1} d_K, \hat{A}_{:K}(\hat{A}_{:K}^T \hat{A}_{:K})^{-1} d_K \rangle = \langle d_K, (\hat{A}_{:K}^T \hat{A}_{:K})^{-1} d_K \rangle > 0.$$

Therefore, it follows from Lemma 4.1 and the facts $\bar{\lambda} > 0$ and $\langle P^K b, h^K \rangle = 0$ that

$$\varphi(\bar{\lambda}) = \sqrt{\|P^K b\|^2 + \bar{\lambda}^2 \|h^K\|^2} > 0,$$

which implies that

$$v^K = \bar{\lambda} \|h^K\|^2 / \varphi(\bar{\lambda}) \in \mathcal{V}(\bar{\lambda}) \quad \text{and} \quad v^K > 0.$$

Since $K \in \mathcal{B}(\bar{\lambda})$ is arbitrarily chosen, we know that $v > 0$ for all $v \in \mathcal{V}(\bar{\lambda})$.

When $p(\cdot) = \|\cdot\|_1$, the set $\partial p(0)$ has the representation of

$$\partial p(0) = \{u \in \mathbb{R}^n \mid -1 \leq u_i \leq 1, i \in [n]\}.$$

In other words, $B = [I_n \quad -I_n]^T \in \mathbb{R}^{2n \times n}$ and $d = e_{2n}$. Therefore, $d_K \neq 0$ for any $\bar{\lambda} \in (0, \lambda_\infty)$ and $K \in \mathcal{B}(\bar{\lambda})$. \square

The numerical results in section 7 will show that the secant method and the semismooth Newton method are comparable for solving the least-squares constrained Lasso problem, which also demonstrates the high efficiency of the secant method even for the case that the HS-Jacobian can be computed.

6. An AS-based secant method for $(\text{CP}(\varrho))$. A main computational challenge (especially for high-dimensional problems) for solving (E_φ) comes from computing the function value of $\varphi(\cdot)$, which requires solving the optimization problem $(\text{PLS}(\lambda))$. To address this challenge, we will incorporate AS to Algorithm 5.2.

6.1. An AS technique for sparse optimization problems. We briefly introduce the AS technique developed in [38] for solving sparse optimization problems of the following form:

$$(6.1) \quad \min_{x \in \mathbb{R}^n} \{\Phi(x) + P(x)\},$$

where $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable convex function and $P : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ is a closed proper convex function. We assume that the convex composite optimization problem (6.1) has at least one solution. Certainly, the optimization problem $(P_{LS}(\lambda))$ is a special case of (6.1). We define the proximal residual function $R : \mathbb{R}^n \rightarrow \mathbb{R}^n$ as

$$(6.2) \quad R(x) := x - \text{Prox}_P(x - \nabla\Phi(x)), \quad x \in \mathbb{R}^n.$$

The norm of $R(x)$ is a standard measurement for the quality of an obtained solution, and x is a solution to (6.1) if and only if $R(x) = 0$.

Let $I \subseteq [n]$ be an index set. We consider the following constrained optimization problem with the index set I :

$$(6.3) \quad \min_{x \in \mathbb{R}^n} \{\Phi(x) + P(x) \mid x_{I^c} = 0\},$$

Algorithm 6.1 An AS strategy for solving (6.1).

- 1: **Input:** an initial index set $I_0 \subseteq [n]$, a given tolerance $\epsilon \geq 0$, and a given positive integer k_{\max} (e.g., $k_{\max} = 500$).
- 2: **Output:** a solution x^* to the problem (6.1) satisfying $\|R(x^*)\| \leq \epsilon$.
- 3: **1.** Find

$$(6.4) \quad x^0 \in \arg \min_{x \in \mathbb{R}^n} \left\{ \Phi(x) + P(x) - \langle \delta^0, x \rangle \mid x_{I_0^c} = 0 \right\},$$

where $\delta^0 \in \mathbb{R}^n$ is an error vector such that $\|\delta^0\| \leq \epsilon$ and $(\delta^0)_{I_0^c} = 0$.

- 2: Compute $R(x^0)$, and set $s = 0$.
- 4: **while** $\|R(x^s)\| > \epsilon$ **do**
- 5: **3.1.** Create J_{s+1} as

$$(6.5) \quad J_{s+1} = \left\{ j \in I_s^c \mid (R(x^s))_j \neq 0 \right\}.$$

If $J_{s+1} = \emptyset$, let $I_{s+1} \leftarrow I_s$; otherwise, let k be a positive integer satisfying $k \leq \min\{|J_{s+1}|, k_{\max}\}$, and define

$$\widehat{J}_{s+1} = \left\{ j \in J_{s+1} \mid |(R(x^s))_j| \text{ is among the first } k \text{ largest values in } \{|(R(x^s))_i|\}_{i \in J_{s+1}} \right\}.$$

Update I_{s+1} as

$$I_{s+1} \leftarrow I_s \cup \widehat{J}_{s+1}.$$

- 6: **3.2.** Solve the constrained problem

$$(6.6) \quad x^{s+1} \in \arg \min_{x \in \mathbb{R}^n} \left\{ \Phi(x) + P(x) - \langle \delta^{s+1}, x \rangle \mid x_{I_{s+1}^c} = 0 \right\},$$

where $\delta^{s+1} \in \mathbb{R}^n$ is an error vector such that $\|\delta^{s+1}\| \leq \epsilon$ and $(\delta^{s+1})_{I_{s+1}^c} = 0$.

- 7: **3.3:** Compute $R(x^{s+1})$, and set $s \leftarrow s + 1$.
 - 8: **end while**
 - 9: **return:** Set $x^* = x^s$.
-

where $I^c = [n] \setminus I$ is the complement of I . A key fact is that a solution to (6.3) is also a solution to (6.1) if there exists a solution \bar{x} to (6.1) such that $\text{supp}(\bar{x}) \subseteq I$. The AS technique is motivated by this fact. Specifically, starting with a reasonable guessing $I_0 \subseteq [n]$, the AS technique is an adaptive strategy to refine the current index set I_k based on a solution to (6.3) with $I = I_k$. We present the details of the AS technique for solving (6.1) in Algorithm 6.1.

It is worthwhile mentioning that in Algorithm 6.1, the error vectors $\delta^0, \{\delta^{s+1}\}$ in (6.4) and (6.6) are not given but imply that the corresponding minimization problems can be solved inexactly. We can just take $\delta^s = 0$ (for $s \geq 0$) if we solve the reduced subproblems exactly. The following proposition shows that we can obtain an inexact solution by solving a reduced problem with a much smaller dimension.

PROPOSITION 6.1 (see [38, Proposition 1]). *For any given nonnegative integer s , the updating rule of x^s in Algorithm 6.1 can be interpreted in the procedure as follows. Let M_s be a linear map from $\mathbb{R}^{|I_s|}$ to \mathbb{R}^n defined as*

$$(M_s z)_{I_s} = z, \quad (M_s z)_{I_s^c} = 0, \quad z \in \mathbb{R}^{|I_s|},$$

and Φ^s, P^s be functions from $\mathbb{R}^{|I_s|}$ to \mathbb{R} defined as $\Phi^s(z) := \Phi(M_s z)$, $P^s(z) := P(M_s z)$ for all $z \in \mathbb{R}^{|I_s|}$. Then $x^s \in \mathbb{R}^n$ can be computed as

$$(x^s)_{I_s} := \text{Prox}_{P^s}(\hat{z} - \nabla \Phi^s(\hat{z})),$$

and $(x^s)_{I_s^c} = 0$, where \hat{z} is an approximate solution to the problem

$$(6.7) \quad \min_{z \in \mathbb{R}^{|I_s|}} \left\{ \Phi^s(z) + P^s(z) \right\},$$

which satisfies

$$(6.8) \quad \|\hat{z} - \text{Prox}_{P^s}(\hat{z} - \nabla \Phi^s(\hat{z})) + \nabla \Phi^s(\text{Prox}_{P^s}(\hat{z} - \nabla \Phi^s(\hat{z}))) - \nabla \Phi^s(\hat{z})\| \leq \epsilon,$$

and ϵ is the parameter given in Algorithm 6.1.

The finite termination property of Algorithm 6.1 for solving (6.1) is shown in the following proposition.

PROPOSITION 6.2 (see [38, Theorem 1]). *For any given initial index set $I_0 \subseteq [n]$ and tolerance $\epsilon \geq 0$, the while loop in Algorithm 6.1 will terminate after a finite number of iterations.*

The high efficiency of the AS technique for solving a wide class of sparse optimization problems in the form of (6.1) has been demonstrated in [38, 37, 15, 36].

6.2. An AS-based secant method for $(\text{CP}(\varrho))$. When applying the level-set method to solve $(\text{CP}(\varrho))$, one needs to solve a sequence of regularized problems in the form of $(\text{P}_{\text{LS}}(\lambda))$ with $\lambda \in \{\lambda_k\}_{k \geq 0}$ generated by the root-finding algorithm for solving (E_φ) . This naturally motivates us to incorporate Algorithm 6.1 into Algorithm 5.2 for solving the regularized least-squares problems in the form of $(\text{P}_{\text{LS}}(\lambda))$ since we can effectively construct an initial index set for the AS technique based on the solution of the previous problem on the solution path. For the first problem on the solution path, we will choose λ_0 to be relatively large such that its solution is highly sparse.

Algorithm 6.2 SMOP: A root-finding-based secant method for $(\text{CP}(\varrho))$.

- 1: **Input:** $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^n$, $\mu \in (0, 1)$, $0 < \underline{\lambda} < \lambda_1 < \lambda_0 \leq \bar{\lambda} \leq \lambda_\infty$ satisfying $\varphi(\underline{\lambda}) < \varrho < \varphi(\bar{\lambda})$.
 - 2: **Step 1.** Call Algorithm 6.1 with $I_0 = \emptyset$ to solve $(\text{P}_{\text{LS}}(\lambda))$ with $\lambda = \lambda_0$, and obtain the solution $x(\lambda_0)$. Compute $\varphi(\lambda_0)$.
 - 3: **while** $\varphi(\lambda_k) \neq \varrho$ **do**
 - 4: **Step 2.1.** Set $k = k + 1$. Set $I_0^k = \{i \in [n] \mid (x(\lambda_{k-1}))_i \neq 0\}$.
 - 5: **Step 2.2.** Call Algorithm 6.1 with $I_0 = I_0^k$ to solve $(\text{P}_{\text{LS}}(\lambda))$ with $\lambda = \lambda_k$ to obtain $x(\lambda_k)$, and compute $\varphi(\lambda_k)$.
 - 6: **Step 2.3.** Generate λ_{k+1} by Algorithm 5.2.
 - 7: **end while**
 - 8: **Step 3.** Set $\lambda^* = \lambda_k$ and $x^* = x(\lambda_k)$.
 - 9: **Return:** λ^* and x^* .
-

In such a way, we choose $I_0 = \emptyset$ in Algorithm 6.1 for the first problem. We present the details in Algorithm 6.2.

We make some remarks before concluding this section. First, we can naturally apply Algorithm 6.2 to efficiently generate a solution path for $(\text{CP}(\varrho))$ with a sequence of noise-level controlling parameters $\varrho_1 > \varrho_2 > \cdots > \varrho_T > 0$. Second, we know that if we apply the AS technique to the level-set method based on (E_φ) , we may easily encounter the infeasibility issue if $\tau_{k+1} < \tau_k$ in (1.1).

7. Numerical experiments. In this section, we will present numerical results to demonstrate the high efficiency of our proposed SMOP. We will focus on solving $(\text{CP}(\varrho))$ with two objective functions: (1) the ℓ_1 penalty: $p(x) = \|x\|_1$, $x \in \mathbb{R}^n$; (2) the sorted ℓ_1 penalty: $p(x) = \sum_{i=1}^n \gamma_i |x|_{(i)}$, $x \in \mathbb{R}^n$, with given parameters $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_n \geq 0$ and $\gamma_1 > 0$, where $|x|_{(1)} \geq |x|_{(2)} \geq \cdots \geq |x|_{(n)}$, which serve as illustrative examples to highlight the efficiency of our algorithm. It is worthwhile mentioning that the sorted ℓ_1 penalty is not separable. For demonstration purposes only, we will test the performance of SMOP when $p(\cdot)$ is a nonpolyhedral function at the end of this section.

In our numerical experiments, we measure the accuracy of the obtained solution \tilde{x} for $(\text{CP}(\varrho))$ by the following relative residual:

$$\eta := \frac{|\tilde{\varphi} - \varrho|}{\max\{1, \varrho\}},$$

where $\tilde{\varphi} := \|A\tilde{x} - b\|$. We test all algorithms on datasets from the UCI Machine Learning Repository as in [16, 17], which are originally obtained from the LIBSVM datasets [7]. Table 3 presents the statistics of the tested UCI instances. All our computational results are obtained using MATLAB R2023a on a Windows workstation with the following specifications: a 12-core Intel Core i7-12700 (2.10GHz) processor and 64GB of RAM. In all the tables presented in this section, $\text{nnz}(x)$ represents the number of elements in the solution x obtained by SMOP (with a stopping tolerance of 10^{-6}) for solving $(\text{CP}(\varrho))$ that have an absolute value greater than 10^{-8} . In addition, we denote BMOP (NMOP) as the root-finding-based bisection method (hybrid of the bisection method and the semismooth Newton method) for solving the optimization problem $(\text{CP}(\varrho))$.

TABLE 3
Statistics of the UCI test instances.

Problem idx	Name	m	n	Sparsity(A)	Norm(b)
1	E2006.train	16087	150360	0.0083	452.8605
2	log1p.E2006.train	16087	4272227	0.0014	452.8605
3	E2006.test	3308	150358	0.0092	221.8758
4	log1p.E2006.test	3308	4272226	0.0016	221.8758
5	pyrim5	74	201376	0.5405	5.7768
6	triazines4	186	635376	0.6569	9.1455
7	bodyfat7	252	116280	1.0000	16.7594
8	housing7	506	77520	1.0000	547.3813

TABLE 4

The values of c to obtain $\varrho = c\|b\|$ for the ℓ_1 penalized problems with least-squares constraints. In this table, $c_{LS} = \frac{\lambda^*}{\|A^T b\|_\infty}$ represents the regularization parameter for the corresponding $P_{LS}(\lambda^*)$, where the optimal solution λ^* to $\varphi(\lambda) = \varrho$ is obtained by SMOP.

Test	idx	1	2	3	4	5	6	7	8
I	c	0.1	0.1	0.08	0.08	0.05	0.1	0.001	0.1
	nnz(x)	339	110	246	405	79	655	107	148
	c_{LS}	2.6-7	2.8-4	4.2-7	2.1-4	5.7-3	2.8-3	1.1-6	1.3-3
II	c	0.09	0.09	0.06	0.06	0.015	0.03	0.0001	0.04
	nnz(x)	1387	1475	884	1196	92	497	231	377
	c_{LS}	1.1-7	6.2-5	1.7-7	9.6-5	3.0-4	5.6-5	3.8-8	3.0-5

7.1. The ℓ_1 penalized problems with least-squares constraints. In this subsection, we focus on the problem $(CP(\varrho))$ with $p(\cdot) = \|\cdot\|_1$. We will compare the efficiency of SMOP to the state-of-the-art SSNAL-LSM algorithm [17], SPGL1 solver [33, 35], and ADMM. Moreover, we perform experiments to demonstrate that our secant method is considerably more efficient than the bisection method for root finding while performing on par with the semismooth Newton method, where the HS-Jacobian is computable.

In practice, we have multiple choices for solving the subproblems in SMOP. In our experiments, we utilized the squared smoothing Newton method [23, 12] and SSNAL to solve the subproblems in SMOP. The maximum number of iterations for SPGL1, SSNAL, and ADMM is set to 100,000, while for SMOP and SSNAL-LSM, the maximum number of iterations of the outermost loop is set to 200. Additionally, we have set the maximum running time to 1 hour. To select ϱ , we use the values of c in Table 4 for each instance listed in Table 3 and let $\varrho = c\|b\|$. Finally, we point out that the AS technique is not employed in the SSNAL-LSM.

We compare SMOP to SPGL1, SSNAL-LSM, and ADMM to solve $(CP(\varrho))$ with the tolerances of 10^{-4} and 10^{-6} , respectively. The test results are presented in Table 5. These results indicate that SMOP successfully solves all the tested instances and outperforms SSNAL-LSM, SPGL1, and ADMM. It can be seen from Table 5 that SMOP can achieve a speedup of up to 1,000 times compared to SPGL1 for the problems that can be solved by SPGL1 (a significant number of instances cannot be solved by SPGL1 to the required accuracy). Regarding ADMM, SMOP remains significantly superior in terms of efficiency for all cases, with a speedup of over 1,300 times. Furthermore, compared to SSNAL-LSM, SMOP also shows vast superiority, with efficiency improvements up to more than 260 times. Note that SPGL1 has two modes: the primal mode (denoted by SPGL1) and the hybrid mode (denoted by SPGL1_H).

TABLE 5

The performance of SMOP (A1), SSNAL-LSM (A2), SPGL1(A3), and ADMM (A4) in solving the ℓ_1 penalized problems with least-squares constraints (CP(ρ)) with $\rho = c\|b\|$, where the specific value of c for each problem is listed in Table 4. In the table, the underline is used to mark cases where the algorithm fails to reach the given tolerance. For simplicity, we omit the “e” in the scientific notation.

idx	Time (s)				η				Outermost iter			
	A1	A2	A3	A4	A1	A2	A3	A4	A1	A2	A3	A4
Test I with stoptol = 10^{-4}												
1	1.39+0	2.18+2	3.51+2	4.22+2	2.3-5	4.9-5	1.0-4	1.0-4	24	29	7342	2049
2	2.29+0	5.12+2	1.45+3	6.84+2	3.1-6	7.8-5	9.0-5	8.7-5	12	16	3445	1470
3	4.02-1	5.83+1	3.21+2	8.87+1	9.4-6	2.6-5	1.0-4	1.0-4	24	30	21094	4918
4	1.59+0	2.06+2	7.19+2	9.90+1	1.2-5	7.3-5	9.5-5	1.3-5	13	15	3174	854
5	2.73-1	1.20+1	9.81+0	5.63+0	6.9-6	5.4-6	7.4-5	2.2-5	6	14	498	273
6	2.32+0	1.74+2	3.35+2	1.01+2	5.8-6	4.4-5	9.1-5	7.5-5	9	17	1987	571
7	4.35-1	9.12+0	8.98+0	8.59+0	2.8-5	5.9-5	9.8-5	9.9-5	15	18	539	583
8	2.99-1	9.07+0	1.29+1	7.94+0	2.6-5	8.6-5	1.0-4	9.0-5	10	14	515	424
Test I with stoptol = 10^{-6}												
1	1.45+0	3.22+2	1.51+3	7.06+2	2.5-7	6.1-8	9.9-7	1.0-6	25	36	28172	3539
2	2.52+0	6.68+2	1.75+3	3.42+3	9.9-8	3.5-8	9.2-7	9.9-7	13	24	4155	8725
3	4.12-1	7.40+1	<u>2.11+3</u>	1.81+2	1.1-8	2.3-7	<u>6.2-6</u>	1.0-6	25	35	<u>100000</u>	10100
4	1.72+0	3.40+2	1.04+3	4.03+2	1.3-9	5.7-7	7.2-7	7.9-7	14	26	4584	3820
5	2.93-1	1.61+1	4.58+1	3.95+2	1.0-7	6.0-8	9.1-7	9.8-7	7	19	2468	20155
6	2.47+0	2.13+2	8.24+2	2.31+3	3.0-7	4.0-7	8.2-7	3.4-7	10	23	5578	13672
7	4.68-1	1.18+1	9.11+0	1.85+1	1.9-9	9.6-7	2.7-7	9.9-7	17	22	544	1250
8	3.28-1	1.45+1	3.84+1	4.40+1	2.4-7	8.4-8	4.0-7	8.7-7	11	24	1539	2427
Test II with stoptol = 10^{-4}												
1	7.26+0	4.51+2	1.38+3	6.12+2	3.0-6	4.6-5	1.0-4	1.0-4	26	30	27775	3014
2	6.79+0	1.54+3	1.32+3	4.01+2	1.8-5	3.6-5	9.7-5	6.8-5	14	21	3000	733
3	3.51+0	1.84+2	<u>1.50+3</u>	1.34+2	1.3-5	2.3-5	<u>8.7-2</u>	1.0-4	25	29	<u>100000</u>	7333
4	2.91+0	6.91+2	6.23+2	4.94+1	7.5-6	3.6-6	9.6-5	5.8-5	14	22	2694	385
5	6.23-1	1.53+1	8.65+0	2.01+1	2.8-5	7.9-6	6.6-5	9.5-5	9	13	395	1000
6	9.02+0	3.46+2	<u>3.60+3</u>	3.82+2	6.8-6	3.7-5	<u>7.6-2</u>	9.9-5	12	17	<u>24924</u>	2232
7	1.50+0	1.59+1	3.06+2	3.39+1	1.6-5	8.7-6	9.9-5	9.8-5	12	18	19820	2340
8	2.37+0	1.90+1	1.69+2	1.19+1	1.4-6	8.9-5	9.1-5	9.8-5	13	18	5914	644
Test II with stoptol = 10^{-6}												
1	7.23+0	5.96+2	<u>3.60+3</u>	8.82+2	3.7-9	2.9-7	<u>3.6-2</u>	1.0-6	27	35	<u>62384</u>	4453
2	7.37+0	1.85+3	2.04+3	1.46+3	1.4-7	3.9-7	9.7-7	1.0-7	15	27	4688	3464
3	3.59+0	2.36+2	<u>1.49+3</u>	1.99+2	8.1-10	8.3-7	<u>8.7-2</u>	1.0-6	26	36	<u>100000</u>	11051
4	3.02+0	8.44+2	1.37+3	2.18+2	3.1-9	4.3-7	9.9-7	6.0-7	15	28	5912	1980
5	6.37-1	2.49+1	4.14+2	1.48+2	2.4-7	3.0-8	8.7-7	9.7-7	10	22	22091	7592
6	9.37+0	4.25+2	<u>3.60+3</u>	3.60+3	5.4-11	6.7-7	<u>7.5-2</u>	1.2-7	14	22	<u>25158</u>	21556
7	1.59+0	2.09+1	3.37+2	8.54+1	3.2-7	1.9-8	8.8-7	9.7-7	13	23	21523	5817
8	2.39+0	2.68+1	1.65+3	3.34+1	4.5-7	6.9-7	8.8-7	9.8-7	14	26	59147	1834

We do not print the results of SPGL1_H since SPGL1 outperforms SPGL1_H in most of the cases in our tests.

Subsequently, we perform numerical experiments to compare the performance of the secant method to the bisection method and the HS-Jacobian-based semismooth Newton method for finding the root of (E_φ) to further illustrate the efficiency of SMOP. Figure 1 presents the ratio of the computation time between BMOP and NMOP to the computation time of SMOP on solving (CP(ρ)) for some instances. The numerical results show that SMOP easily beats BMOP, with a large margin when a higher-precision solution is required. The results also reveal that SMOP performs

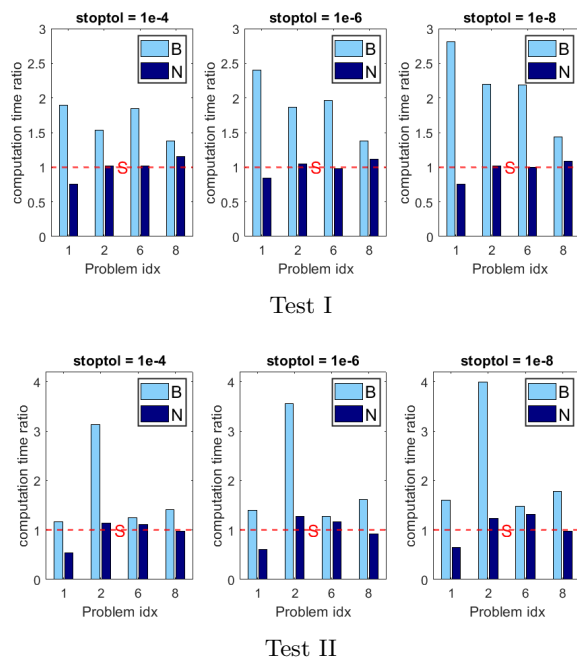


FIG. 1. The ratio of the computation time between BMOP (B) and NMOP (N) to the computation time of SMOP in solving $(CP(\varrho))$ with the ℓ_1 regularization.

comparably to NMOP for solving the ℓ_1 penalized least-squares constrained problem, in which the HS-Jacobian is computable. This is more strong evidence to illustrate the efficiency of SMOP.

Next, we perform tests on BMOP and SMOP to generate a solution path for $(CP(\varrho))$ involving multiple choices of tuning parameters $\varrho > 0$. In this test, we solve $(CP(\varrho))$ with $\varrho_i = c_i \cdot c \|b\|$, $i = 1, \dots, 100$, where $c_i = 1.5 - 0.5 \times (i - 1)/99$ and c is the same constant as in Table 4. In this test, we apply the warm-start strategy to both algorithms. The average iteration numbers of BMOP and SMOP and the ratio of the computation time of BMOP to the computation time of SMOP are shown in Figure 2 with a tolerance of 10^{-6} . From this figure, it is evident that utilizing the secant method for root finding significantly reduces the number of iterations by around 4 times. The reduction in iterations results in a substantial decrease in computation time for SMOP, which is typically less than one-third of the time required by BMOP.

7.2. The sorted ℓ_1 penalized problems with least-squares constraints.

In this subsection, we will present the numerical results of SMOP in solving the sorted ℓ_1 penalized problems with least-squares constraints $(CP(\varrho))$. For comparison purposes, we also conducted tests on Newt-ALM-LSM (similar to SSNAL-LSM but with the subproblems solved by Newt-ALM [18]) and ADMM for $(CP(\varrho))$.

In our numerical experiments, we choose the parameters $\gamma_i = 1 - (i - 1)/(n - 1)$, $i = 1, \dots, n$, in the sorted ℓ_1 penalty function $p(x) = \sum_{i=1}^n \gamma_i |x_{(i)}|$, $x \in \mathbb{R}^n$. The maximum iteration number is set to 200 for SMOP and Newt-ALM-LSM and 100,000 for ADMM. The subproblems in SMOP are solved by Newt-ALM in this test. In all experiments within this subsection, the stopping tolerance is set to 10^{-6} . The results obtained with a tolerance of 10^{-4} are similar to those obtained with a tolerance of 10^{-6} ; therefore,

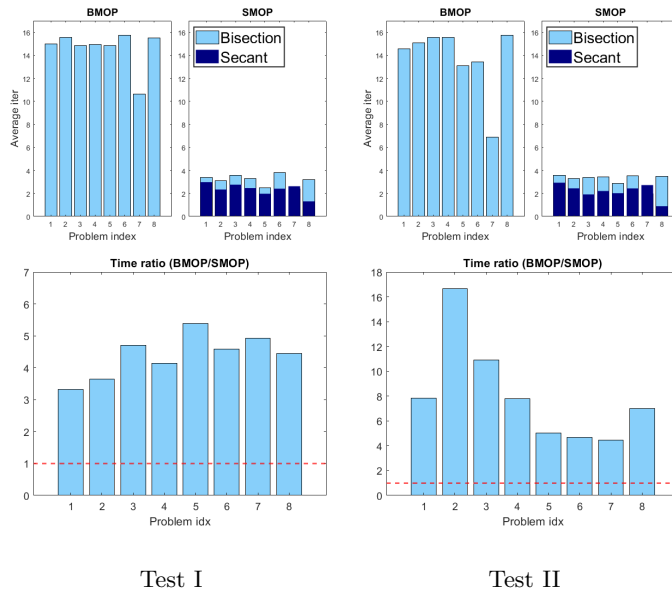


FIG. 2. The performance of BMOP and SMOP in generating a solution path for $(CP(\rho))$ with the ℓ_1 regularization and stopping tolerance 10^{-6} .

TABLE 6

The performance of SMOP (A1), Newt-ALM-LSM (A2), and ADMM (A4) in solving the sorted ℓ_1 penalized problems with least-squares constraints $(CP(\rho))$ with $\rho = c\|b\|$. In the table, $c_{LS} = \frac{\lambda^*}{\|A^T b\|_\infty}$ represents the regularization parameter for the corresponding $P_{LS}(\lambda^*)$, where the optimal solution λ^* to $\varphi(\lambda) = \rho$ is obtained by SMOP. The stopping tolerance is set to 10^{-6} , and the underline is used to mark cases where the algorithm fails to reach the given tolerance. For simplicity, we omit the “e” in the scientific notation.

idx	c	nnz(x)	c_{LS}	Time (s)			η			Outermost iter		
				A1	A2	A4	A1	A2	A4	A1	A2	A4
Test I												
2	0.15	3	2.4-2	3.84+0	1.34+2	<u>3.60+3</u>	1.1-7	5.3-7	<u>2.8-1</u>	8	21	<u>8637</u>
4	0.1	3	4.8-3	4.79+0	1.35+2	<u>3.60+3</u>	6.0-7	8.9-7	<u>2.9-4</u>	10	17	<u>28891</u>
5	0.1	113	1.9-2	6.29-1	4.98+1	4.23+2	1.0-7	4.5-7	1.5-7	7	22	17974
6	0.15	413	1.0-2	3.10+0	2.43+2	<u>3.60+3</u>	2.7-7	1.6-7	<u>1.9-4</u>	9	21	<u>19071</u>
7	0.002	22	1.9-5	3.56-1	1.67+1	2.44+1	3.6-9	6.0-7	9.9-7	14	22	1616
8	0.15	95	6.9-3	6.06-1	2.55+1	1.57+2	1.3-7	7.7-7	9.0-7	10	23	8329
Test II												
1	0.1	339	2.6-7	2.53+1	1.40+2	5.13+2	2.9-7	5.6-7	1.0-6	25	34	2490
2	0.095	629	1.0-4	5.39+1	4.82+2	2.87+3	1.7-7	2.9-7	9.4-7	17	27	6770
3	0.08	246	4.2-7	4.98+0	6.54+1	1.60+2	2.0-8	7.1-7	1.0-6	25	36	8491
4	0.07	758	1.4-4	2.26+1	4.26+2	5.86+2	4.0-8	9.0-7	9.8-7	16	27	4550
5	0.02	95	5.7-4	2.05+0	9.87+1	3.58+2	3.2-8	5.6-7	7.6-7	11	20	15582
6	0.05	997	5.5-4	2.32+1	1.04+3	<u>3.60+3</u>	8.4-7	2.1-7	<u>3.5-6</u>	10	23	<u>19159</u>
7	0.001	107	1.1-6	1.02+0	2.85+1	1.30+1	5.9-8	6.9-9	9.5-7	17	22	826
8	0.08	206	4.3-4	3.38+0	1.03+2	5.58+1	5.7-9	7.4-7	3.8-7	13	25	2842

we will not present them in order to save space. We set $\rho = c\|b\|$, where the values of c are specified in Table 6. The numerical results are presented in Table 6. From the table, it is evident that SMOP outperforms Newt-ALM-LSM and ADMM for all

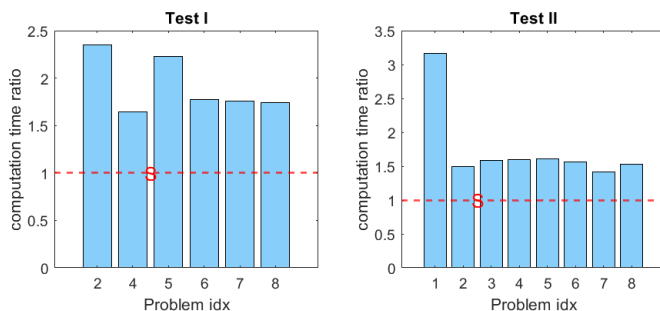


FIG. 3. The ratio of the computation time between BMOP to the computation time of SMOP in solving $(CP(\varrho))$ with the sorted ℓ_1 regularization.

the cases. More specifically, SMOP can be up to around 80 times faster than Newt-ALM-LSM and up to more than 600 times faster than ADMM for the problems that can be solved by ADMM. Additionally, Figure 3 presents the computation time ratio between BMOP and SMOP for both Test I and Test II. This also demonstrates the significance of the secant method in root finding for achieving higher efficiency.

7.3. A group Lasso penalized problems with least-squares constraints.

In this subsection, we will present the numerical experiments conducted to solve a group Lasso penalized problems with least-squares constraints. The purpose of this demonstration is to illustrate the potential and high efficiency of our proposed secant method in solving the equation $\varphi(\lambda) = \varrho$ for the nonpolyhedral function penalized problems with least-squares constraints. We will compare our algorithm, SMOP, with other state-of-the-art algorithms to demonstrate its high efficiency and robustness.

We consider the following penalty function $p(\cdot)$ in this subsection:

$$(7.1) \quad p(x) = \sum_{t=1}^l \sqrt{x_{2t-1}^2 + x_{2t}^2}, \quad x \in \mathbb{R}^{2l}.$$

For the purpose of demonstration, we will keep using the UCI dataset that was utilized in the previous two subsections. However, it is necessary to ensure that the value of n is even. Next, we group the i th and $(i+1)$ th elements together for all $i = 1, 3, \dots, n-1$. The values of c utilized to obtain $\varrho = c \|b\|$ are presented in Table 7. In SMOP, the subproblems are solved by SSNAL [39]. The maximum iteration number for both SMOP and SSNAL-LSM is set to 200, while for SPGL1 and ADMM, their maximum iteration number is set to 100,000. As for the maximum running time, it remains set at 1 hour. Next, we will compare SMOP with the state-of-the-art algorithms SSNAL-LSM, SPGL1, and ADMM. The results of the tests are presented in Table 8. From the table, it is evident that SMOP outperforms SSNAL-LSM, SPGL1, and ADMM with speedups of up to 300, 900, and 1,100, respectively. In addition, Figure 4 illustrates the ratio of computation time between BMOP and SMOP. This figure clearly shows that using the secant method can greatly enhance overall efficiency, resulting in a speed improvement of approximately 1.5–4 times, even when dealing with the nonpolyhedral penalty function (7.1).

8. Conclusion. In this paper, we have designed an efficient sieving-based secant method for solving $(CP(\varrho))$. When $p(\cdot)$ is a polyhedral gauge function, we have proven that for any $\bar{\lambda} \in (0, \lambda_\infty)$, all $v \in \partial\varphi(\bar{\lambda})$ are positive. Consequently, when

TABLE 7

The values of c to obtain $\rho = c\|b\|$ for the group Lasso penalized problems with least-squares constraints. In the table, $c_{LS} = \frac{\lambda^*}{\|A^T b\|_\infty}$ represents the regularization parameter for the corresponding $P_{LS}(\lambda^*)$, where the optimal solution λ^* to $\varphi(\lambda) = \rho$ is obtained by SMOP.

	idx	c	nnz(x)	c_{LS}
Test I	4	0.1	6	4.4-3
	5	0.1	50	2.4-2
	6	0.15	138	1.3-2
	7	0.002	28	2.4-5
	8	0.15	66	8.4-3
Test II	1	0.105	95	7.5-7
	3	0.08	403	4.3-7
	4	0.08	731	2.2-4
	5	0.02	120	9.1-4
	6	0.05	372	6.3-4
	7	0.001	186	1.3-6
	8	0.08	260	4.9-4

TABLE 8

The performance of SMOP (A1), SSNAL-LSM (A2), SPGL1 (A3), and ADMM (A4) in solving the group Lasso penalized problems with least-squares constraints (CP(ρ)) with $\rho = c\|b\|$. The stopping tolerance is set to 10^{-6} , and the underline is used to mark cases where the algorithm fails to reach the given tolerance. For simplicity, we omit the “e” in the scientific notation.

idx	Time (s)				η				Outermost iter			
	A1	A2	A3	A4	A1	A2	A3	A4	A1	A2	A3	A4
Test I												
4	3.75+0	1.16+2	8.49+2	<u>3.60+3</u>	1.3-7	3.1-7	6.37-7	<u>7.2-5</u>	11	21	3024	<u>22125</u>
5	8.14-1	2.74+2	2.96+1	9.16+2	1.4-9	3.5-7	6.05-7	1.0-6	11	21	1319	38530
6	5.19+0	1.46+3	1.70+2	3.02+3	3.2-10	4.5-7	5.98-7	9.8-7	10	22	1086	15768
7	5.98-1	8.80+0	3.02+1	2.59+1	3.7-8	5.0-7	2.07-7	1.0-6	14	19	2102	1627
8	6.88-1	1.41+2	8.30+0	1.19+2	1.8-8	2.6-7	2.46-7	9.6-7	9	22	334	6211
Test II												
1	3.29+0	4.33+1	3.18+3	1.12+3	2.7-7	2.7-7	9.8-7	1.0-6	24	29	55596	5826
3	3.83+0	3.00+1	<u>2.06+3</u>	2.57+2	1.3-7	3.4-7	<u>3.8-6</u>	1.0-6	22	36	<u>100000</u>	13031
4	2.97+1	2.42+3	1.19+3	5.86+2	5.2-7	9.6-9	8.6-7	7.4-7	13	27	4241	3401
5	1.70+0	1.29+2	3.30+2	9.27+1	8.0-7	1.7-8	8.9-7	6.5-7	9	20	18001	3959
6	2.51+1	1.39+3	<u>3.60+3</u>	3.60+3	1.3-8	1.4-7	<u>5.8-5</u>	2.6-7	11	22	<u>20646</u>	19075
7	1.22+0	1.88+1	5.99+2	2.69+1	5.3-8	2.1-8	6.9-7	9.9-7	15	23	41578	1685
8	5.75+0	1.47+2	1.14+2	1.94+2	2.5-7	3.7-7	4.4-7	9.8-7	15	25	4373	9974

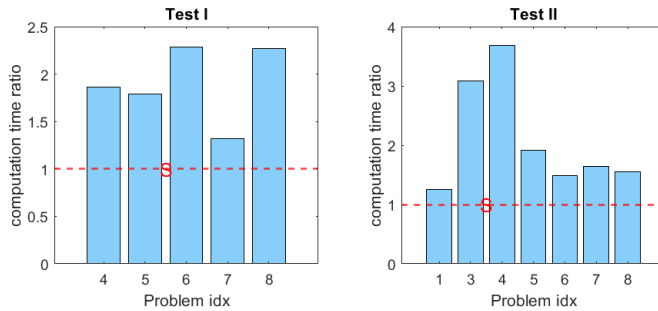


FIG. 4. The ratio of the computation time between BMOP to the computation time of SMOP in solving (CP(ρ)) with the group Lasso regularization.

$p(\cdot)$ is a polyhedral gauge function, the secant method can solve (E_φ) with at least a 3-step Q-quadratic convergence rate. We have demonstrated the high efficiency of our method for solving $(CP(\varrho))$ by two representative instances, specifically, the ℓ_1 and the sorted ℓ_1 penalized constrained problems. It is worth mentioning that to the best of our knowledge, calculating $\partial_{\text{HS}}\varphi(\cdot)$ or $\partial\varphi(\cdot)$ is not an easy task for the sorted ℓ_1 penalized constrained problems. Moreover, our numerical results on the ℓ_1 penalized constrained problems, in which the $\partial_{\text{HS}}\varphi(\cdot)$ is computable as shown in Proposition 5.6, have verified that the efficiency of SMOP is not compromised compared to the performance of the HS-Jacobian-based semismooth Newton method. This motivates us to use the secant method instead of the semismooth Newton method for solving (E_φ) regardless of the availability of the generalized Jacobians. For future research, we will investigate the properties of $\varphi(\cdot)$ for nonpolyhedral functions $p(\cdot)$, particularly when the nondegenerate condition does not hold.

Acknowledgments. The authors would like to thank the referees and the associate editor for their valuable suggestions to improve the quality of this paper. Thanks also go to Mr. Jiaming Ma at The Hong Kong Polytechnic University for his helpful discussions on the proof of Proposition 3.3(ii).

REFERENCES

- [1] A. Y. ARAVKIN, J. V. BURKE, D. DRUSVYATSKIY, M. P. FRIEDLANDER, AND S. ROY, *Level-set methods for convex optimization*, Math. Program., 174 (2019), pp. 359–390.
- [2] A. BECK AND M. TEOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.
- [3] J. BOLTE, A. DANILIDIS, AND A. LEWIS, *Tame functions are semismooth*, Math. Program., 117 (2009), pp. 5–19.
- [4] J. F. BONNANS AND A. SHAPIRO, *Perturbation Analysis of Optimization Problems*, Springer Series in Operations Research, Springer-Verlag, Berlin, 2000.
- [5] Z. X. CHAN AND D. F. SUN, *Constraint nondegeneracy, strong regularity, and nonsingularity in semidefinite programming*, SIAM J. Optim., 19 (2008), pp. 370–396.
- [6] X. D. CHEN, D. F. SUN, AND J. SUN, *Complementarity functions and numerical experiments on some smoothing Newton methods for second-order-cone complementarity problems*, Comput. Optim. Appl., 25 (2003), pp. 39–56.
- [7] C. CHIH-CHUNG, *LIBSVM: A library for support vector machines*, ACM Trans. Intell. Syst. Technol., 2 (2011), pp. 1–27.
- [8] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.
- [9] F. FACCHINEI AND J.-S. PANG, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer Series in Operations Research, Springer-Verlag, Berlin, 2003.
- [10] M. P. FRIEDLANDER AND P. TSENG, *Exact regularization of convex programs*, SIAM J. Optim., 18 (2007), pp. 1326–1350.
- [11] D. GABAY AND B. MERCIER, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Comput. Math. Appl., 2 (1976), pp. 17–40.
- [12] Y. GAO AND D. F. SUN, *Calibrating least squares covariance matrix problems with equality and inequality constraints*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 1432–1457.
- [13] R. GLOWINSKI AND A. MARROCO, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires*, Revue française d'automatique, informatique, recherche opérationnelle, Analyse numérique, 9 (1975), pp. 41–76.
- [14] J. HAN AND D. F. SUN, *Newton and quasi-Newton methods for normal maps with polyhedral sets*, J. Optim. Theory Appl., 94 (1997), pp. 659–676.
- [15] Q. LI, B. JIANG, AND D. F. SUN, *MARS: A second-order reduction algorithm for high-dimensional sparse precision matrices estimation*, J. Mach. Learn. Res., 24 (2023), pp. 1–44.
- [16] X. LI, D. F. SUN, AND K.-C. TOH, *A highly efficient semismooth Newton augmented Lagrangian method for solving Lasso problems*, SIAM J. Optim., 28 (2018), pp. 433–458.
- [17] X. LI, D. F. SUN, AND K.-C. TOH, *On efficiently solving the subproblems of a level-set method for fused Lasso problems*, SIAM J. Optim., 28 (2018), pp. 1842–1866.

- [18] Z. LUO, D. F. SUN, K.-C. TOH, AND N. XIU, *Solving the OSCAR and SLOPE models using a semismooth Newton-based augmented Lagrangian method*, J. Mach. Learn. Res., 20 (2019), pp. 1–25.
- [19] F. MENG, D. F. SUN, AND G. ZHAO, *Semismoothness of solutions to generalized equations and the Moreau-Yosida regularization*, Math. Program., 104 (2005), pp. 561–581.
- [20] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [21] F. A. POTRA, *On Q-order and R-order of convergence*, J. Optim. Theory Appl., 63 (1989), pp. 415–431.
- [22] F. A. POTRA, L. QI, AND D. F. SUN, *Secant methods for semismooth equations*, Numer. Math., 80 (1998), pp. 305–324.
- [23] L. QI, D. F. SUN, AND G. ZHOU, *A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities*, Math. Program., 87 (2000), pp. 1–35.
- [24] L. QI AND P. TSENG, *On almost smooth functions and piecewise smooth functions*, Nonlinear Anal. Theory Methods Appl., 67 (2007), pp. 773–794.
- [25] B. RECHT, M. FAZEL, AND P. A. PARRILO, *Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization*, SIAM Rev., 52 (2010), pp. 471–501.
- [26] H. E. RICHARD AND C. M. GEORGE, *Computer Algebra Recipes: A Gourmet’s Guide to the Mathematical Models of Science*, Springer-Verlag, Berlin, 2001.
- [27] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [28] D. F. SUN, *A further result on an implicit function theorem for locally Lipschitz functions*, Oper. Res. Lett., 28 (2001), pp. 193–198.
- [29] D. F. SUN, *The strong second-order sufficient condition and constraint nondegeneracy in nonlinear semidefinite programming and their implications*, Math. Oper. Res., 31 (2006), pp. 761–776.
- [30] D. F. SUN AND J. SUN, *Semismooth matrix-valued functions*, Math. Oper. Res., 27 (2002), pp. 150–169.
- [31] R. TIBSHIRANI, M. SAUNDERS, S. ROSSET, J. ZHU, AND K. KNIGHT, *Sparsity and smoothness via the fused Lasso*, J. R. Stat. Soc. Ser. B Stat. Methodol., 67 (2005), pp. 91–108.
- [32] J. F. TRAUB, *Iterative Methods for the Solution of Equations*, Prentice Hall, Englewood Cliffs, NJ, 1964.
- [33] E. VAN DEN BERG AND M. P. FRIEDLANDER, *Probing the Pareto frontier for basis pursuit solutions*, SIAM J. Sci. Comput., 31 (2008), pp. 890–912.
- [34] E. VAN DEN BERG AND M. P. FRIEDLANDER, *Sparse optimization with least-squares constraints*, SIAM J. Optim., 21 (2011), pp. 1201–1229.
- [35] E. VAN DEN BERG AND M. P. FRIEDLANDER, *SPGL1: A Solver for Large-Scale Sparse Reconstruction*, 2019, <https://friedlander.io/spgl1>.
- [36] C. WU, Y. CUI, D. LI, AND D. F. SUN, *Convex and nonconvex risk based statistical learning at scale*, INFORMS J. Comput., (2023).
- [37] Y. YUAN, T.-H. CHANG, D. F. SUN, AND K.-C. TOH, *A dimension reduction technique for large-scale structured sparse optimization problems with application to convex clustering*, SIAM J. Optim., 32 (2022), pp. 2294–2318.
- [38] Y. YUAN, M. LIN, D. F. SUN, AND K.-C. TOH, *Adaptive Sieving: A Dimension Reduction Technique for Sparse Optimization Problems*, arXiv:2306.17369, 2023.
- [39] Y. ZHANG, N. ZHANG, D. F. SUN, AND K.-C. TOH, *An efficient Hessian based algorithm for solving large-scale sparse group Lasso problems*, Math. Program., 179 (2020), pp. 223–263.