A superlinearly convergent subgradient method for sharp semismooth problems

Vasileios Charisopoulos[*] Damek Davis[†]

January 12, 2022

Abstract

Subgradient methods comprise a fundamental class of nonsmooth optimization algorithms. Classical results show that certain subgradient methods converge sublinearly for general Lipschitz convex functions and converge linearly for convex functions that grow sharply away from solutions. Recent work has moreover extended these results to certain nonconvex problems. In this work we seek to improve the complexity of these algorithms, asking: is it possible to design a superlinearly convergent subgradient method? We provide a positive answer to this question for a broad class of sharp semismooth functions.

1 Introduction

Subgradient methods are a popular class of nonsmooth optimization algorithms for minimizing locally Lipschitz functions $f\colon \mathbb{R}^d \to \mathbb{R}$:

$$\operatorname{minimize}_{x\in\mathbb{R}^d} f(x).$$

Given an initial iterate $x_0 \in \mathbb{R}^d$, the basic method repeats

$$x_{k+1} = x_k - \alpha_k v_k \qquad \text{for } v_k \in \partial f(x_k),$$

where $\{\alpha_k\}$ is a control sequence and $\partial f(x)$ denotes the *Clarke subdifferential* at a point $x \in \mathbb{R}^d$, comprised of limiting convex combinations of gradients at nearby points [63]. While the method originated over fifty years ago in convex optimization [26, 30, 58, 59, 66] (with later extensions to nonconvex problems [16, 27, 51, 52, 53]), it has recently become a popular and successful technique both in modern deep learning problems (e.g., in Google's

[*] School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14853, USA; `people.orie.cornell.edu/vc333/`

[†] School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14850, USA; `people.orie.cornell.edu/dsd95/`. Research of Davis supported by an Alfred P. Sloan research fellowship and NSF DMS award 2047637.

Tensorflow [1]) and in robust low-rank matrix estimation problems [10]. For the latter problem class, recent work has highlighted the prevalence and benefits of the so-called *sharp growth* property, which stipulates that f grows at least linearly away from its minimizers:

$$f(x) - \inf f \geq \mu \cdot \operatorname{dist}(x, \mathcal{X}_*),$$

where $\mathcal{X}_* = \operatorname{argmin} f$. For convex problems (and more generally *weakly convex* problems), this classical regularity condition leads to local linear convergence provided the sequence $\{\alpha_k\}$ is chosen appropriately (see also [17, 26, 30, 37, 58, 66, 67, 76]). While linear convergence is desirable, we ask:

> Is it possible to design a locally superlinearly convergent subgradient method?

In this paper, we design such a method for a wide class of *sharp* and *semismooth* problems.

Setting the stage, assume for simplicity that f has a unique minimizer $\bar{x}$ and optimal value 0. The starting point for our method is the classical subgradient method with Polyak stepsize, which iterates

$$x_{k+1} = x_k - \frac{f(x_k)}{\|v_k\|^2} v_k, \qquad \text{where } v_k \in \partial f(x_k).$$

This method converges linearly for sharp convex [58] and weakly convex [17] problems and admits the following reformulation:

$$x_{k+1} = \operatorname*{argmin}_{x \in \mathbb{R}^d} \|x - x_k\|^2 \qquad \text{subject to: } f(x_k) + \langle v_k, x - x_k \rangle \leq 0. \tag{1}$$

Seeking to improve the linear convergence of (1), a natural strategy proposed in Polyak's original work [58] is to augment the constraint (1) with a collection $\{(y_i, v_i)\}_{i=1}^n$ of points y_i and subgradients $v_i \in \partial f(y_i)$, resulting in the update:

$$x_{k+1} = \operatorname*{argmin}_{x \in \mathbb{R}^d} \|x - x_k\|^2 \qquad \text{subject to: } [f(y_i) + \langle v_i, x - y_i \rangle]_{i=1}^n \leq 0, \tag{2}$$

The work [58] suggests choosing y_i among $\{x_j\}_{j \leq k}$ and shows that the iterates x_k converge linearly for Lipschitz convex functions (similar to (1)). While there is no theoretical convergence rate improvement, the work [58] suggests the method (2) improves upon (1) numerically, though the per-iteration cost may grow substantially if n is large.

A strategy akin to (2) also appears in the literature on so-called *bundle methods* [44, 75]. Instead of aggregating inequalities as in [58], these methods build piecewise linear models of the objective function and output the proximal point of the models. If the proximal point sufficiently decreases the objective, the algorithm takes a "serious step." Otherwise, the algorithm takes a "null step," which consists of using subgradient information to improve the model. Bundle methods often perform well in practice and their convergence/complexity theory is understood in several settings [23, 25, 33, 38, 39, 47, 54, 64]. Most relevantly for this work, on sharp convex functions, variants of the bundle method converge superlinearly relative to the number of serious steps [50] and converge linearly relative to both serious and null steps [18].

In this work, we study a slight variant of the update (2), where the "$\leq$" is replaced by an equality and the points y_i are chosen iteratively. This variant is motivated by our second assumption – semismoothness. In short, semismoothness ensures that $\bar{x}$ is nearly feasible for the equation $f(x_k) + \langle v, x - x_k\rangle = 0$ when x_k is near $\bar{x}$ and $v \in \partial f(x_k)$. More formally, the function f is *semismooth* at $\bar{x}$ [49] whenever

$$f(x) + \langle v, \bar{x} - x\rangle = o(\|\bar{x} - x\|) \qquad \text{as } x \to \bar{x} \text{ and } v \in \partial f(x), \tag{3}$$

where $o(\cdot)$ is any univariate function satisfying $\lim_{t\to 0} o(t)/t = 0$. While it may at first seem stringent, semismoothness is a reasonable assumption since it holds for any locally Lipschitz weakly convex [49] or semialgebraic function [6].

Turning to our main algorithm, we depart from the quadratic programming problem of (2) and instead construct both our iterates x_k and the collection $\{(y_i, v_i)\}_i$ by solving a sequence of linear systems, a simpler operation in general. At iteration k, we construct the collection as follows: set initial point $y_0 = x_k$, choose subgradient $v_0 \in \partial f(y_0)$, and for $j = 1, \ldots, d$, recursively set

$$y_j := \operatorname*{argmin}_{x \in \mathbb{R}^d} \|x - x_k\|^2 \qquad \text{subject to: } [f(y_i) + \langle v_i, x - y_i\rangle]_{i=0}^{j-1} = 0 \tag{4}$$

and choose $v_j \in \partial f(y_j)$ arbitrarily. For this collection, we will show that the next iterate

$$x_{k+1} \in \operatorname*{argmin}_{y \in \{y_i\}_i} f(y) \qquad \text{satisfies} \qquad f(x_{k+1}) = o(f(x_k)) \quad \text{as } k \to \infty.$$

The construction of y_i may at first seem mysterious, but its success results from a simple "lemma of alternatives" proved in this work. Namely, suppose that the first $j-1$ elements $y_1, \ldots, y_{j-1}$ do not superlinearly improve on x_k. Then we prove that one of the following must hold: either y_j superlinearly improves upon x_k or the rank of $[v_i^\mathsf{T}]_{i=0}^{j}$ is $j+1$. In this way we must obtain local superlinear improvement in at most d steps.

Thus, for sharp semismooth functions, simply repeating (4) will result in superlinear convergence in a small, dimension dependent neighborhood of $\bar{x}$. While this method converges superlinearly, its theoretical region of admissible initializers is small. Our numerical experiments suggest this may be a limitation of the analysis, rather than of the algorithm. Nevertheless, it is desirable to have a linearly convergent fallback method that quickly reaches the region of superlinear convergence from a much larger set of initial conditions. To that end, we extend the linear convergence of the Polyak subgradient method (1) to sharp and semismooth functions (see Theorem 2.1). The argument and result mirror the previous result for weakly convex functions [17].

While the Polyak algorithm eventually reaches the region of superlinear convergence, its entrance may be hard to detect. Thus, we provide a generic procedure for coupling the superlinear steps (4) with the Polyak algorithm (1) (or another fallback algorithm), which rapidly converges to the region of superlinear convergence when initialized in a much larger region. The coupled algorithm may be implemented with knowledge of a single parameter, namely, the optimal value $f(\bar{x})$. An intriguing open problem, left to future work, is whether one can design a parameter free variant.

The results stated thus far assume that $\mathcal{X}_*$ is isolated at $\bar{x}$. We prove that all of the algorithms analyzed in this work converge superlinearly to nonisolated solutions for functions that are (b)-*regular* along $\mathcal{X}_*$, a natural uniformization of the semismoothness property that was recently analyzed in [15]. We review and provide several examples of the (b)-regularity property and develop a calculus for creating further examples, going beyond the setting of [15]. For example, we show that a composition $f = h \circ F$ is (b)-regular along $\mathcal{X}_*$ whenever (i) F is a smooth mapping and (ii) h is a locally Lipschitz semialgebraic function with isolated minimum $\bar{y} \in \mathrm{Range}(F)$. We use these results to provide useful corollaries for root-finding and feasibility problems and discuss relations to the literature on semismooth Newton methods [29, 36, 40, 42, 61, 62] and accelerations of projection methods [55, 56].

Finally, we note that despite local superlinear convergence, the worst-case complexity of the proposed method depends on d, a property not in line with the "dimension free" complexity theory of first-order methods. Nevertheless, we found that we may terminate (4) early in several scenarios, yielding promising empirical performance. For example, in Figure 1 we plot the performance of the proposed method, dubbed `SuperPolyak`, against the method (1), dubbed `PolyakSGM`, on a simple low-rank matrix sensing problem. Here the problem of interest is simply

$$f(U, V) := \frac{1}{m} \left\| \mathcal{A}(UV^{\mathsf{T}}) - \mathcal{A}(\bar{M}) \right\|_1 \qquad \text{for all } U, V \in \mathbb{R}^{d \times r},$$

where $\bar{M} \in \mathbb{R}^{d \times d}$ is a fixed rank r matrix and $\mathcal{A} : \mathbb{R}^{d \times d} \to \mathbb{R}^m$ is a linear operator; see Section 5.2.1 for a more detailed description. From the the plots, we see the proposed method performs well in terms of time and oracle complexity and appears less sensitive to the condition number $\tilde{\kappa}$ of the matrix $\bar{M}$. Beyond early termination, we also introduce and use several other implementation strategies, including one that reduces the naive arithmetic complexity cost of constructing the points y_i from $O(d^4)$ (ignoring subgradient evaluations) to $O(d^3)$ arithmetic operations. With these strategies in place, the advantage of `SuperPolyak` persists in several scenarios outlined in our numerical illustration.

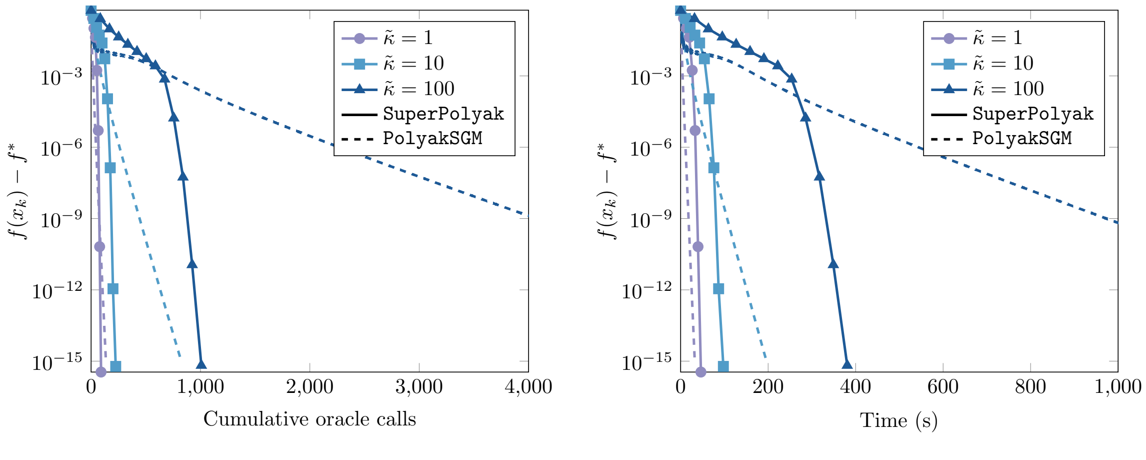

Figure 1: Low-rank matrix sensing with Hadamard measurements, varying condition number $\tilde{\kappa}$, and parameters $d = 2^{15}$, $r = 2$ and $m = 16d$. See Section 5.2.1 for description.

Before turning to the formal statements of the results, the following section formalizes the basic notations and constructions used throughout this work.

1.1 Notation and basic constructions

We will mostly follow standard notation used in convex analysis as set out in the monograph [63]. Throughout, the symbol $\mathbb{R}^d$ will denote a d-dimensional Euclidean space with the inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\|x\| = \sqrt{\langle x, x\rangle}$. We denote the open ball of radius $\varepsilon > 0$ around a point $x \in \mathbb{R}^d$ by the symbol $B_\varepsilon(x)$. We use the symbol $\bar{B}$ to denote the closed unit ball at the origin. A set-valued mapping $G \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^m$ maps points $x \in \mathbb{R}^d$ to sets $G(x) \subseteq \mathbb{R}^m$. We say a set-valued mapping G is nonempty-valued if $G(x)$ is nonempty for every $x \in \mathbb{R}^d$ and locally bounded if $G(\mathcal{X}) := \bigcup_{x\in\mathcal{X}} G(x)$ is a bounded set for any bounded set $\mathcal{X} \subseteq \mathbb{R}^d$. For any set $\mathcal{X} \subseteq \mathbb{R}^d$, the *distance function* and the *projection map* are defined by

$$\operatorname{dist}(x, \mathcal{X}) := \inf_{y\in\mathcal{X}} \|y - x\| \qquad \text{and} \qquad P_{\mathcal{X}}(x) := \operatorname*{argmin}_{y\in\mathcal{X}} \|y - x\|,$$

respectively. Given a function $f \colon \mathbb{R}^d \to \mathbb{R}$ and $\gamma > 0$, we define the proximal operator $\operatorname{prox}_{\gamma f} \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^m$ of f to be the set-valued mapping with values:

$$\operatorname{prox}_{\gamma f}(x) := \operatorname*{argmin}_{y} \left\{ f(y) + \frac{1}{2\gamma}\|y - x\|^2 \right\} \qquad \text{for all } x \in \mathbb{R}^d.$$

We call a function $h \colon \mathbb{R}^d \to \mathbb{R}$ *sublinear* if its epigraph is a closed convex cone, and in that case we define

$$\operatorname{Lin}(h) = \{x \in \mathbb{R}^d \colon h(x) = -h(-x)\}$$

to be its *lineality space.* Given a matrix A, we denote its spectral norm by $\|A\|_2$.

Semialgebraicity. We call a set $\mathcal{X} \subseteq \mathbb{R}^d$ *semialgebraic* if it is the union of finitely many sets defined by finitely many polynomial inequalities. Likewise, we call a function $f \colon \mathbb{R}^d \to \mathbb{R}$ semialgebraic if its graph $\operatorname{gph}(f) = \{(x, f(x)) \colon x \in \mathbb{R}^d\}$ is semialgebraic. Finally, we call a set-valued mapping $G \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^m$ *semialgebraic* if its graph $\operatorname{gph}(G) = \{(x, y) \colon y \in G(x)\}$ is semialgebraic.

Subdifferentials. Consider a locally Lipschitz function $f \colon \mathbb{R}^d \to \mathbb{R}$ and a point x. The Clarke subdifferential is the convex hull of limits of gradients evaluated at nearby points

$$\partial f(x) = \operatorname{conv}\left\{ \lim_{i\to\infty} \nabla f(x_i) \colon x_i \xrightarrow{\Omega} x \right\},$$

where $\Omega \subseteq \mathbb{R}^d$ is the set of points at which f is differentiable (recall Radamacher's theorem). If f is L-Lipschitz on a neighborhood U, then for all $x \in U$ and $v \in \partial f(x)$, we have $\|v\| \leq L$. A point $\bar{x}$ satisfying $0 \in \partial f(x)$ is said to be critical for f. A function f is called *ρ-weakly convex* on an open convex set U if the perturbed function $f + \frac{\rho}{2}\|\cdot\|^2$ is convex on U. The Clarke subgradients of such functions automatically satisfy the uniform approximation property:

$$f(y) \geq f(x) + \langle v, y - x \rangle - \frac{\rho}{2}\|y - x\|^2 \qquad \text{for all } x, y \in U, v \in \partial f(x).$$

Finally consider a locally Lipschitz mapping $F \colon \mathbb{R}^d \to \mathbb{R}^m$. Then the *Clarke Jacobian of F at x* is the set

$$\partial F(x) = \operatorname{conv}\left\{ \lim_{i \to \infty} \nabla F(x_i) \colon x_i \xrightarrow{\Omega} x \right\},$$

where $\Omega \subseteq \mathbb{R}^d$ is the set of points at which F is differentiable.

Normal cones. Let $\mathcal{X}$ be a closed set and let $\bar{x} \in \mathcal{X}$. The *Fréchet normal cone* to $\mathcal{X}$ at $\bar{x}$, denoted by $N_{\mathcal{X}}^F(\bar{x})$, consists of all vectors $v \in \mathbb{R}^d$ satisfying

$$\langle v, x - \bar{x} \rangle \leq o(\|x - \bar{x}\|) \qquad \text{as } x \xrightarrow{\mathcal{X}} \bar{x}.$$

The *Limiting normal cone* to $\mathcal{X}$ at $\bar{x}$, denoted by $N_{\mathcal{X}}^L(\bar{x})$, consists of all vectors $v \in \mathbb{R}^d$ such that there exist sequences $x_i \in \mathcal{X}$ and $v_i \in N_{\mathcal{X}}^F(x_i)$ satisfying

$$(x_i, v_i) \to (x, v) \text{ as } i \to \infty.$$

The *Clarke normal cone* of $\mathcal{X}$ at x, denoted by $N_{\mathcal{X}}(x)$, consists of all convex combinations of limiting normal vectors

$$N_{\mathcal{X}}(x) = \operatorname{cl}\operatorname{conv} N_{\mathcal{X}}^L(\bar{x}).$$

The normal cone is related to the distance function as follows:

$$\partial \operatorname{dist}(x, \mathcal{X}) = \begin{cases} \operatorname{conv} \frac{x - P_{\mathcal{X}}(x)}{\operatorname{dist}(x, \mathcal{X})} & \text{if } x \notin \mathcal{X}; \\ N_{\mathcal{X}}(x) \cap \bar{B} & \text{otherwise.} \end{cases} \tag{5}$$

Finally, we recall that whenever $x \notin \mathcal{X}$ and $\hat{x} \in P_{\mathcal{X}}(x)$, we have $\frac{x - \hat{x}}{\|x - \hat{x}\|} \in N_{\mathcal{X}}(\hat{x})$.

Manifolds. We will need a few basic results about smooth manifolds, which can be found in the references [7, 43]. A set $\mathcal{M} \subseteq \mathbb{R}^d$ is called a C^p smooth manifold (with $p \geq 1$) if there exists a natural number m, an open neighborhood U of x, and a C^p smooth mapping $F \colon U \to \mathbb{R}^m$ such that the Jacobian $\nabla F(x)$ is surjective and $\mathcal{M} \cap U = F^{-1}(0)$. The tangent and normal spaces to $\mathcal{M}$ at $x \in \mathcal{M}$ are defined to be $T_{\mathcal{M}}(x) = \ker(\nabla F(x))$ and $N_{\mathcal{M}}(x) = T_{\mathcal{M}}(x)^{\perp} = \operatorname{Range}(\nabla F(x)^*)$, respectively. If $\mathcal{M}$ is a C^2-smooth manifold around a point $\bar{x}$, then there exists $C > 0$ such that $y - x \in T_{\mathcal{M}}(x) + C\|y - x\|^2 \bar{B}$ for all $x, y \in \mathcal{M}$ near $\bar{x}$.

2 Assumptions, algorithms, and main results

In this section, we introduce our assumptions, algorithms, and main results. To that end, throughout this work we consider the problem

$$\text{minimize}_{x\in\mathbb{R}^d}\ f(x), \tag{6}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a locally Lipschitz function with optimal value f^*. We denote $\mathcal{X}_* = \operatorname{argmin}_{x\in\mathbb{R}^d} f(x)$ and assume that $\mathcal{X}_* \neq \emptyset$. We also fix a point $\bar{x} \in \mathcal{X}_*$ and a radius $\delta > 0$, which factor into our initialization assumptions.

2.1 Main assumptions: sharpness and (b)-regularity

In this section, we formalize our assumptions on the growth and semismoothness of f. We additionally provide three concrete example problem classes.

2.1.1 Sharp growth

Our first technical assumption is that f grows sharply away from $\mathcal{X}_*$:

(A1) **(Sharpness)** There exists $\mu > 0$ such that the estimate

$$f(x) - f^* \geq \mu \operatorname{dist}(x, \mathcal{X}_*) \qquad \text{holds for all } x \in B_\delta(\bar{x}).$$

Assumption (A1) is a classical regularity condition known to ensure (local) linear convergence of subgradient methods in the (weakly) convex setting [17]. Sharp growth is known to hold in a range of problems, most classically in feasibility formulations of linear programs [34] (see also the survey [57]). Several contemporary problems also exhibit sharp growth, for example, nonconvex formulations of low-rank matrix sensing and completion problems [10].

2.1.2 Semismoothness

Our second technical assumption is that f satisfies a "uniform semismoothness" condition with respect to $\mathcal{X}_*$:

(A2) **((b)-regularity)** There exists a locally bounded nonempty-valued set-valued mapping $g\colon \mathbb{R}^d \rightrightarrows \mathbb{R}^d$ and constants $C_{(b)}, \eta > 0$ such that the estimate

$$|f(x) + \langle v, y - x\rangle - f^*| \leq C_{(b)} \|y - x\|^{1+\eta} \tag{7}$$

holds for all $x \in B_\delta(\bar{x})$, $v \in g(x)$, and $y \in \mathcal{X}_* \cap B_{2\delta}(\bar{x})$.

Assumption (A2) is a uniformization of the classical semismoothness property (3) of [49]. In particular, the estimate (7) is identical to (3) when $\mathcal{X} = \{\bar{x}\}$, $g = \partial f$, and the term $\|\cdot\|^{1+\eta}$ is replaced with any univariate function $o(\cdot)$ satisfying $\lim_{t\to 0} o(t)/t = 0$. We require the stronger error modulus $\|\cdot\|^{1+\eta}$ to deal with $\mathcal{X}_*$ that are not singleton sets.

Importantly if $\mathcal{X}_*$ is a singleton all results of this work easily generalize to "little-o" error. The recent work [15] introduced the general (b)-regularity estimate, provided several basic examples, and developed a calculus, focusing on the mapping $g = \partial f$. In Section 3, we recall and extend the results of [15], introducing new examples and proving a formal chain rule for (b)-regularity. The latter result ensures that g computed by certain automatic differentiation schemes are valid generalized gradient mappings [5].

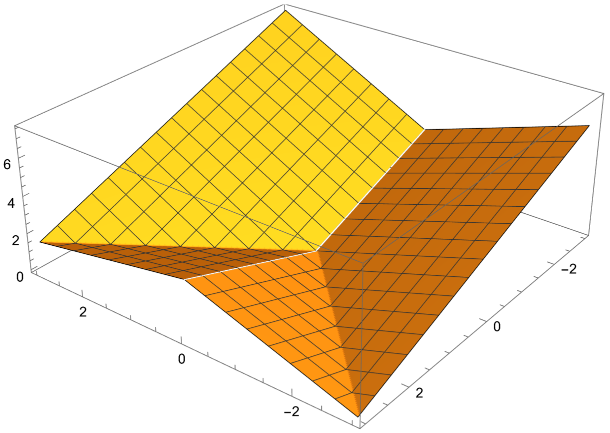

Figure 2: The function $f(x, y) = |y - |x|| + \max(x, 0)$ satisfies Assumptions (A1) and (A2) along the set $\mathcal{X}_* = \{(x, -x)) \colon x \leq 0\}$.

2.1.3 Examples

We now provide three concrete problem classes where (b)-regularity holds. At the end of the section, we also touch upon sharp growth. We present the proofs of all three Propositions in Section 3.4.

The first class arises from semialgebraic functions composed with smooth mappings.

Proposition 2.1. *Consider a C^2 smooth mapping $F \colon \mathbb{R}^d \to \mathbb{R}^m$ and a locally Lipschitz semialgebraic function $h \colon \mathbb{R}^m \to \mathbb{R}$. Suppose that $y \in \mathbb{R}^m$ is an isolated minimum of h and define $\mathcal{X}_* = F^{-1}(y)$. Then for any $\bar{x} \in \mathcal{X}_*$, the function*

$$f(x) = h(F(x)) \qquad \textit{for all } x \in \mathbb{R}^d,$$

satisfies Assumption (A2) *along $\mathcal{X}_*$ at $\bar{x}$ with mapping g defined by the formal chain rule:*

$$g(x) = \nabla F(x)^{\mathsf{T}} \partial h(F(x)) \qquad \textit{for all } x \in \mathbb{R}^d.$$

A second class of examples arises from root finding problems.

Proposition 2.2. *Consider a C^2 smooth mapping $F_1 \colon \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and a locally Lipschitz semialgebraic mapping $F_2 \colon \mathbb{R}^{d_2} \to \mathbb{R}^{d_3}$. Define $F := F_2 \circ F_1$ and the set $\mathcal{X}_* := F^{-1}(0)$. Then for any $\bar{x} \in \mathcal{X}_*$ at which $F_2(\bar{x})$ is an isolated zero of F_1, the function*

$$f(x) = \|F(x)\| \qquad \textit{for all } x \in \mathbb{R}^d,$$

satisfies Assumption (A2) *along $\mathcal{X}_*$ at $\bar{x}$ with mapping g defined by the formal chain rule:*

$$g(x) = \begin{cases} \nabla F_1(x)^\mathsf{T} \partial F_2(x)^\mathsf{T} \frac{F(x)}{\|F(x)\|} & F(x) \neq 0; \\ \nabla F_1(x)^\mathsf{T} \partial F_2(x)^\mathsf{T} \bar{B} & \text{otherwise;} \end{cases} \qquad \text{for all } x \in \mathbb{R}^d,$$

where $\partial F_2(x)^\mathsf{T}$ denotes the set of transposed elements of the Clarke Jacobian of F_2 at x.

Finally we present a class arising in feasibility problems.

Proposition 2.3. *Consider a collection of semialgebraic sets $\mathcal{X}_i \subseteq \mathbb{R}^d$ indexed by a finite set I. Suppose that $\bigcap_{i \in I} \mathcal{X}_i = \{\bar{x}\}$ and define $\mathcal{X}_* := \{\bar{x}\}$. Then for any $\bar{x} \in \mathcal{X}_*$, the function*

$$f(x) = \sum_{i \in I} \operatorname{dist}(x, \mathcal{X}_i) \qquad \text{for all } x \in \mathbb{R}^d,$$

satisfies Assumption (A2) *along $\mathcal{X}_*$ at $\bar{x}$ with mapping g defined by the formal sum rule:*

$$g(x) = \sum_{i \in I} \partial \operatorname{dist}(x, \mathcal{X}_i) \qquad \text{for all } x \in \mathbb{R}^d.$$

To close this section, we mention that the sharp growth property (A1) is well-studied in the settings of these propositions. For example, the setting of Proposition 2.1 arises in low-rank matrix estimation problems [10], where regularity property (A1) is a consequence of the *restricted isometry property* [9] of the "measurement operator." Next, for f defined in Proposition 2.2, regularity property (A1) is simply the classical *metric subregularity assumption.* This is a weak regularity property known to hold in many circumstances [35, 57]. Finally, for f defined in Proposition 2.3, regularity property (A1) is simply the classical *linear regularity assumption*, which is known to ensure local linear convergence of the alternating projection method for closed sets [20, Theorem 3.2.3]. The property is automatic, for example, for intersections of convex polyhedral sets (see [4, Fact 5.8]), and moreover holds for "generic perturbations" of semialgebraic sets [22, Theorem 7.1]. We present further analysis of these settings in Section 2.5.

2.2 `PolyakSGM`: local linear convergence

We now turn to the first method of this work, dubbed `PolyakSGM`, which is shown in Algorithm 1. This method will be a key subroutine in the locally superlinearly convergent algorithm developed in Section 2.4.

Algorithm 1 `PolyakSGM`(z_0, ϵ)

```
repeat for $i = 0, 1, \ldots$
    Choose $v_i \in g(z_i)$
    if $v_i = 0$ then
        return $z_i$
    end if
    $z_{i+1} := z_i - \frac{f(z_i) - f^*}{\|v_i\|^2} v_i$
until $f(z_{i+1}) - f^* \leq \epsilon$
return $z_{i+1}$
```

The following is our main convergence theorem. We place the proof in Section 4.1. We note that the argument mirrors the proof of the analogous result in the convex and weakly convex settings [17, 58].

Theorem 2.1. *Suppose assumptions* (A1) *and* (A2) *hold at* $\bar{x}$. *Let* L *be an upper bound for the maximal norm element of* $g(B_\delta(\bar{x}))$. *Define*

$$\kappa := \frac{L}{\mu} \quad \textit{and} \quad \rho := \sqrt{1 - (2\kappa)^{-2}}.$$

Fix an initial point $x \in \mathbb{R}^d$ *satisfying the bounds:*

$$\|x - \bar{x}\| < \frac{(1-\rho)\delta}{2} \quad \textit{and} \quad \operatorname{dist}(x, \mathcal{X}_*) \leq \left(\frac{\mu}{4C_{(b)}}\right)^{1/\eta}.$$

Then for all $\epsilon > 0$, `PolyakSGM`(x, ϵ) *successfully terminates with at most*

$$\left\lceil 8\kappa^2 \log\left(\frac{\kappa(f(x) - f^*)}{\epsilon}\right)\right\rceil$$

evaluations of g.

We note that it is possible to prove a similar theorem when the (b)-regularity estimate (7) is replaced by the following weaker condition: for some $\gamma < \mu/2$, we have

$$f(x) + \langle v, y - x\rangle - f^* \leq \gamma\|y - x\| \tag{8}$$

for all $x \in B_\delta(\bar{x})$ and $y \in \mathcal{X}_* \cap B_{2\delta}(\bar{x})$. We do not pursue this result since the stronger (b)-regularity estimate (7) will be crucial in what follows.

2.3 `PolyakBundle`: superlinear improvement

In this section, we formally describe the procedure outlined in Equation (4) of the introduction. Specifically, we will show that the `PolyakBundle` procedure shown in Algorithm 2 locally results in superlinear improvement. Note that pseudoinverse computations of Algorithm 2 are identical to the subproblems in (4), but for ease of implementation, we have written the closed-form solution.

Algorithm 2 `PolyakBundle`(x, τ)

```
$y_0 := x$; $v_0 \in g(y_0)$; $A_1 := v_0^\mathsf{T}$.
for $i = 1, \ldots, d$ do
    $y_i = y_0 - A_i^\dagger \begin{bmatrix} f(y_0) - f^* + \langle v_0, y_0 - y_0\rangle \\ \vdots \\ f(y_{i-1}) - f^* + \langle v_{i-1}, y_0 - y_{i-1}\rangle \end{bmatrix}$
    $A_{i+1} := \begin{bmatrix} A_i \\ v_i^\mathsf{T} \end{bmatrix}$ for arbitrary $v_i \in g(y_i)$.
end for
return $y_s$, where $s = \mathrm{argmin}_{i:\|y_i - y_0\| \leq \tau f(y_0)} f(y_i)$
```

Now we turn to our main theorem, which states that the procedure `PolyakBundle` locally results in superlinear improvement. The proof appears in Section 4.2.

Theorem 2.2 (Superlinear Improvement)**.** *Suppose Assumptions* (A1) *and* (A2) *hold at $\bar{x}$. Let L be an upper bound for the maximal norm element of $g(B_\delta(\bar{x}))$ and a Lipschitz constant of f on $B_\delta(\bar{x})$. Then there exists a constant $C_\mathsf{s} > 0$ such that for all scalars $\tau > (3/\mu)$ and points $x \in \mathbb{R}^d$ with*

$$\|x - \bar{x}\| < \frac{\delta}{4}, \qquad \text{and} \qquad \operatorname{dist}(x, \mathcal{X}_*) \leq \min\left\{\left(\frac{\mu}{2C_{(b)}}\right)^{1/\eta}, \left(\frac{\mu^{1-\eta}}{LC_\mathsf{s}}\right)^{1/\eta}\right\},$$

the point $\tilde{x} =$ `PolyakBundle`(x, τ) satisfies

$$f(\tilde{x}) \leq C_\mathsf{s} f(x)^{1+\eta}. \tag{9}$$

We comment on two aspects of this theorem. First, we mention that the requirement that $\|\tilde{x} - y_0\| \leq \tau f(y_0)$ is not necessary for one step of superlinear improvement. However, in Section 2.4 we apply `PolyakBundle` repeatedly and use this condition to ensure $\tilde{x}$ remains near $\bar{x}$. Second, upon checking the proof, the reader will find that the constant C_s can be extremely large, yielding a small region of superlinear convergence:

$$C_\mathsf{s} = O\left(d \max\left\{(L/\mu), L\right\} (8\sqrt{2}L/\mu)^d\right).$$

However, the numerical experiments in Section 5 suggest that this bound may be an artifact of the proof technique. Whether this constant can be improved is an intriguing open question. In Section 2.4 we develop a procedure that reaches the region of superlinear convergence from a more reasonable initial guess, using a reasonable number of evaluations of g. After it reaches this region, the method reverts to `PolyakBundle`.

Remark 1. We mention that a naive implementation of `PolyakBundle` requires $O(d^4)$ operations. In Section 5.1, we develop a strategy that reduces this cost to $O(d^3)$.

2.4 `SuperPolyak`: `PolyakBundle` with a fallback algorithm

In Section 2.3, we showed that the `PolyakBundle`(x, τ) procedure results in superlinear improvement if $\|x - \bar{x}\| \leq \delta/4$ and $\operatorname{dist}(x, \mathcal{X}_*)$ is small. While the former condition is reasonable, the latter appears difficult to satisfy. Thus, in this section, we develop a strategy for coupling the `PolyakBundle` procedure with a linearly convergent fallback algorithm, which rapidly approaches $\mathcal{X}_*$ from a more reasonable initialization.

2.4.1 Fallback algorithms

The method `PolyakSGM` may always be used as a fallback method. However, an alternative fallback method may be preferable to `PolyakSGM`. Two settings of interest arise from fixed-point and feasibility problems.

Example 2.1 (Fixed-point problems)**.** Suppose we seek a fixed-point $\bar{x}$ of a locally Lipschitz mapping $T \colon \mathbb{R}^d \to \mathbb{R}^d$. Then, under the conditions outlined in Proposition 2.2, one may apply `PolyakSGM` to the function $f(x) = \|x - Tx\|$. In place of `PolyakSGM`, one may instead use the classical fixed-point iteration [3, 41, 48], which repeats

$$z_{i+1} = T(z_i).$$

Example 2.2 (Feasibility problems)**.** Suppose we seek a point $\bar{x}$ in the intersection of two closed subsets $\mathcal{X}_1$ and $\mathcal{X}_2$ of $\mathbb{R}^d$. Then under the conditions outlined in Proposition 2.3, one may apply `PolyakSGM` to the function $f(x) = \operatorname{dist}(x, \mathcal{X}_1) + \operatorname{dist}(x, \mathcal{X}_2)$. In place of `PolyakSGM`, one may instead use the classical method of alternating projections [71], which repeats

$$\tilde{z}_i \in P_{\mathcal{X}_1}(z_i); \qquad z_{i+1} \in P_{\mathcal{X}_2}(\tilde{z}_i);$$

Although other fallback methods may be appropriate, we limit our study to algorithms which iterate algorithmic mappings of the following form:

(A3) **(Algorithmic mapping)** There exists radii $0 < \varphi_2 \leq \varphi_1$, a contraction factor $\rho \in (0, 1)$, and a mapping $\mathcal{A} \colon B_{\varphi_1}(\bar{x}) \to \mathbb{R}^d$ such that if $x \notin \mathcal{X}_*$ satisfies

$$\|x - \bar{x}\| < \varphi_1 \qquad \text{and} \qquad \operatorname{dist}(x, \mathcal{X}_*) < \varphi_2,$$

then the following holds:

$$\|\mathcal{A}(x) - \hat{x}\| \leq \rho \operatorname{dist}(x, \mathcal{X}_*) \quad \text{for all } \hat{x} \in P_{\mathcal{X}_*}(x).$$

We call such mappings $\mathcal{A}$ *algorithmic mappings.* For example, we will later show in Lemma 4.4 that `PolyakSGM` is generated by iterating an algorithmic mapping. In the context of Example 2.1, the operator $\mathcal{A} := T$ is an algorithmic mapping if it behaves like a contraction towards points in $\mathcal{X}_*$. Finally, in the context of Example 2.2, [20, Theorem 3.2.3] shows that any selection $\mathcal{A}$ of the set-valued mapping $P_{\mathcal{X}_2} \circ P_{\mathcal{X}_1}$ is an algorithmic mapping provided the sets $\mathcal{X}_1$ and $\mathcal{X}_2$ intersect "transversely" at $\bar{x}$, a property that implies sharp growth of f (see [22] for discussion).

Now consider the iterates $z_0 := \mathcal{A}^{\circ k}(z_0)$ generated by repeatedly applying an algorithmic mapping, starting from some initial point z_0 that is sufficiently close to $\bar{x}$. Then it is straightforward to check that the iterates z^k linearly converge $\mathcal{X}_*$; we provide a simple proof in Lemma 4.3 of Section 4. Thus, any such algorithmic mapping $\mathcal{A}$ generates a well-defined algorithm `FallbackAlg`$(\mathcal{A}, z_0, \epsilon)$, which arises from simply iterating $\mathcal{A}$ until the function gap is of size at most ϵ (see Algorithm 3). Such fallback methods play a key role in our main algorithm, which we now describe.

Algorithm 3 `FallbackAlg`$(\mathcal{A}, z_0, \epsilon)$

```
repeat for $i = 0, 1, \dots$
    $z_{i+1} := \mathcal{A}(z_i)$
until $f(z_{i+1}) - f^* \leq \epsilon$
return $z_{i+1}$
```

2.4.2 Algorithm and main convergence theorem

We now have all the pieces to describe Algorithm 4, which we dub `SuperPolyak`. The method couples `PolyakBundle` and `FallbackAlg`. At each iteration it first attempts a superlinear step. If the step halves the function gap, the method updates the iterate. Otherwise, the method calls the fallback algorithm, which will halve the function gap. Key to the algorithm is the scalar $(3/2)^k$ in line 2, which is eventually larger than $3/\mu$: according to Theorem 2.2, this ensures the `PolyakBundle`$(x_k, (3/2)^k)$ locally results in superlinear improvement. Finally, we mention that one may adjust the performance of the algorithm by changing the factor $(3/2)^k$ or adjusting the constant 1/2 in line 6. We discuss these strategies in Section 5.1 below.

Algorithm 4 `SuperPolyak`$(\mathcal{A}, x_0, \epsilon)$

```
 1: for $k = 0, 1, \dots$ do
 2:     $\tilde{x} :=$ PolyakBundle$(x_k, (3/2)^k)$                         ▷ $\tilde{x}$ can be $\emptyset$
 3:     if $\tilde{x} \neq \emptyset$ and $f(\tilde{x}) - f^* < \frac{1}{2}(f(x_k) - f^*)$ then
 4:         $x_{k+1} := \tilde{x}$                                ▷ PolyakBundle step successful
 5:     else
 6:         $x_{k+1} :=$ FallbackAlg$\left(\mathcal{A}, x_k, \frac{1}{2}(f(x_k) - f^*)\right)$   ▷ Run until function gap halved
 7:     end if
 8:     if $f(x_{k+1}) - f^* \leq \epsilon$ then
 9:         return $x_{k+1}$
10:     end if
11: end for
```

The following theorem shows Algorithm 4 eventually results in superlinear improvement. We place the proof in Section 4.3.

Theorem 2.4. *Suppose Assumptions* (A1), (A2), *and* (A3) *hold at* $\bar{x}$ *for some algorithmic mapping* $\mathcal{A}$ *with contraction factor* ρ *and radii* φ_1 *and* φ_2. *Let* L *be an upper bound for the maximal norm element of* $g(B_\delta(\bar{x}))$ *and a Lipschitz constant of* f *on* $B_\delta(\bar{x})$. *Define*

$$\kappa := \frac{L}{\mu}; \qquad \Delta := f(x_0) - f^*.$$

Fix an initial point $x \in \mathbb{R}^d$ *satisfying the bounds:*

$$\begin{aligned} \|x_0 - \bar{x}\| &\leq \left\{ \left(\frac{2}{1-\rho} \right) \left(1 + \max\left\{ 2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3} \right\} \right) \right\}^{-1} \min\left\{ \frac{\delta}{4}, \varphi_1 \right\}; \\ \operatorname{dist}(x_0, \mathcal{X}_*) &\leq \frac{\varphi_2}{1 + \max\left\{ 2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3} \right\}}. \end{aligned} \tag{10}$$

Define the constant (where C_{s} *appears in Theorem 2.2)*

$$K_1 := \left\lceil \max\left\{ \log_2\left(\Delta \cdot \max\left\{ 2(2C_{\mathsf{s}})^{1/\eta}, \frac{(2C_{(b)})^{1/\eta}}{\mu^{1+1/\eta}}, (\kappa C_{\mathsf{s}})^{1/\eta} \right\} \right), \log_{\frac{3}{2}}\left(\frac{3}{\mu} \right) \right\} \right\rceil. \tag{11}$$

Then for any $\epsilon > 0$, *Algorithm 4 successfully terminates with at most*

1. $\left\lceil \frac{1}{1-\rho} \log(2\kappa) \right\rceil K_1$ *evaluations of* $\mathcal{A}$;
2. $dK_1 + d \left\lceil \frac{\log\log_2(1/\epsilon)}{\log(1+\eta)} \right\rceil$ *evaluations of* g.

The following corollary examines the complexity of Algorithm 4 when the fallback method arises from the Polyak subgradient method. In this setting, evaluating $\mathcal{A}$ requires evaluating both G and f once. We place the proof the following corollary in Section 4.4.

Corollary 2.3. *Consider the setting of Theorem 2.4. Suppose that the fallback algorithm is* `PolyakSGM`(x, ϵ). *Define* $\rho := \sqrt{1 - (2\kappa)^{-2}}$ *and suppose that*

$$\begin{aligned} \|x_0 - \bar{x}\| &\leq \left\{ \left(\frac{2}{1-\rho} \right) \left(1 + \max\left\{ 2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3} \right\} \right) \right\}^{-1} \frac{\delta}{4}; \\ \operatorname{dist}(x_0, \mathcal{X}_*) &\leq \frac{1}{1 + \max\left\{ 2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3} \right\}} \left(\frac{\mu}{4C_{(b)}} \right)^{1/\eta}. \end{aligned}$$

Then Algorithm 4 will successfully terminate after at most

$$\max\left\{ d, \lceil 8\kappa^2 \log(2\kappa) \rceil \right\} K_1 + d \left\lceil \frac{\log\log_2\left(\frac{1}{\epsilon}\right)}{\log(1+\eta)} \right\rceil$$

evaluations of g, *where* K_1 *appears in* (11).

We now turn our attention to further consequences of Theorem 2.4.

2.5 Consequences for root-finding and feasibility problems

In this section, we describe consequences of Theorem 2.4 for root-finding and feasibility problems – two settings where the optimal value f^* is known and equal to zero. For both problem classes, we consider a simple scenario and discuss related literature. Further extensions are possible. For example, we may consider more complex problem structure using the calculus results of the upcoming Section 3. We may also use further generalized gradient maps g. We omit these extensions for brevity.

2.5.1 Root-finding problems

We have the following corollary for root-finding problems. We place the proof in Section 4.5.

Corollary 2.4. *Let $F\colon \mathbb{R}^d \to \mathbb{R}^m$ be a locally Lipschitz mapping. Define $\mathcal{X}_* = F^{-1}(0)$ and let $\bar{x} \in \mathcal{X}_*$. Fix $\mu, \eta > 0$ and assume that*

1. *F is μ-metrically subregular at $\bar{x}$, meaning*

$$\|F(x)\| \geq \mu \operatorname{dist}(x, \mathcal{X}_*) \qquad \textit{for all } x \textit{ near } \bar{x}.$$

2. *$(F, \partial F)$ is (b)-regular along $\mathcal{X}_*$ at $\bar{x}$ with exponent $1+\eta$, meaning there exists $C > 0$ such that the estimate*

$$|F(x) + A(y - x)| \leq C\|y - x\|^{1+\eta}$$

holds for all x near $\bar{x}$, $A \in \partial F(x)$, and $y \in \mathcal{M}$ near $\bar{x}$.

In particular, Item 2 is automatically satisfied when $\mathcal{X}_*$ is isolated at $\bar{x}$ and F is semialgebraic. Now define a function and generalized gradient mapping: for all $x \in \mathbb{R}^d$,

$$f(x) := \|F(x)\| \qquad \textit{and} \qquad g(x) := \begin{cases} \partial F(x)^\mathsf{T} \frac{F(x)}{\|F(x)\|}; \\ \partial F(x)^\mathsf{T} \bar{B}. \end{cases}$$

Then f and g satisfy assumptions (A1) and (A2). Therefore, Algorithm 4 with fallback method `PolyakSGM` locally superlinearly converges to a root of F.

We now place this result in the context of the so-called “semismooth Newton” method, which has a vast literature, summarized in the seminal papers and monographs [29, 36, 40, 42, 61, 62]. To focus our discussion, we compare and contrast Corollary 2.4 with the results of [62]. The semismooth Newton method of [62] directly generalizes the classical Newton method to nonsmooth equations, replacing the classical Jacobian with an element of the Clarke Jacobian. For simplicity we describe this method for square systems $F = 0$, where $F\colon \mathbb{R}^d \to \mathbb{R}^d$ is a locally Lipschitz mapping. To solve this equation, the pioneering work of Qi and Sun [62] considers the following assumptions near a root $\bar{x}$:

1. **(Invertibility)** every $A \in \partial F(\bar{x})$ is invertible (in particular $\bar{x}$ is isolated).

2. **(Semismoothness)** F is semismooth at $\bar{x}$, meaning

$$|F(x) + A(\bar{x} - x)| = o(\|\bar{x} - x\|) \qquad \text{for all } A \in \partial F(x) \text{ as } x \to \bar{x}.$$

Under these assumptions, the work [62] shows that the semismooth Newton iteration

$$x_{k+1} = x_k - A_k^{-1} F(x_k) \qquad \text{for some } A_k \in \partial F(x_k). \tag{12}$$

is locally well-defined and the iterates x_k converge superlinearly to $\bar{x}$. Much work on semismooth Newton methods considers similar conditions to the work of Qi and Sun [62]. While semismoothness is in some sense minimal, it is desirable to weaken the invertibility condition to the metric subregularity condition of Corollary 2.4. Such a result would be useful for the acceleration of certain first-order methods for signal recovery, which may be represented by the fixed-point iteration of Example 2.1. In particular, it is known that the *proximal gradient operator* associated to certain compressive sensing problems [8, 19] is metrically subregular, but does not satisfy the stronger invertibility condition (see Section 5.2.4 for a description of the problem). To the best of our knowledge, Corollary 2.4 presents the first semismooth Newton-type method that converges under the metric subregularity condition, even in the case of an isolated solution of a general semismooth mapping F.

Finally, we mention two further semismooth Newton-type methods that succeed under the metric subregularity condition, but require further assumptions. First, the SuperMann scheme of [68] proposes a nonsmooth (quasi) Newton scheme that converges superlinearly under semi-differentiability and metric subregularity if certain inverse Hessian approximations remain bounded throughout the developed algorithm; the latter property is nontrivial and not verified in that work. Second, the LP-Newton method of [28] proposes a Newton-type methods that converges superlinearly under metric subregularity if a certain smoothness assumption holds; the assumption appears stronger than the classical semismoothness assumption considered in this work.

2.5.2 Feasibility problems

We have the following corollary for feasibility problems. We place the proof in Section 4.6.

Corollary 2.5. *Consider a collection of closed sets $\mathcal{X}_i \subseteq \mathbb{R}^d$ indexed by a finite set I. Define $\mathcal{X}_* := \bigcap_{i \in I} \mathcal{X}_i$ and let $\bar{x} \in \mathcal{X}_*$. Fix $\mu, \eta, C > 0$ and suppose that*

1. *The family $\{\mathcal{X}_i\}_i$ is μ-linearly regular at $\bar{x}$, meaning*

$$\sum_{i \in I} \operatorname{dist}(x, \mathcal{X}_i) \geq \mu \operatorname{dist}(x, \mathcal{X}_*) \qquad \textit{for all } x \textit{ near } \bar{x}.$$

2. *For all $i \in I$, we have*

$$|\langle v, y - x \rangle| \leq C \|v\| \|y - x\|^{1+\eta},$$

for all $x \in \mathcal{X}_i$ and $y \in \mathcal{X}_*$ near $\bar{x}$ and all $v \in N_{\mathcal{X}_i}(x)$.

In particular, Item 2 is automatically satisfied when either (i) $\mathcal{X}_i$ is a C^2 manifold for all $i \in I$ or (ii) $\mathcal{X}_*$ is isolated at $\bar{x}$ and $\mathcal{X}_i$ is semialgebraic or a C^2 smooth manifold around $\bar{x}$ for $i \in I$. Now define a function and generalized gradient mapping: for all $x \in \mathbb{R}^d$,

$$f(x) = \sum_{i \in I} \operatorname{dist}(x, \mathcal{X}_i) \qquad \textit{and} \qquad g(x) = \sum_{i \in I} \partial \operatorname{dist}(x, \mathcal{X}_i),$$

Then f and g satisfy assumptions (A1) *and* (A2)*. Therefore, Algorithm 4 with fallback method* `PolyakSGM` *locally superlinearly converges to $\mathcal{X}_*$.*

Some comments are in order. We note that Item 2 is a natural notion of (b)-regularity for nested sets $\mathcal{X}_* \subseteq \mathcal{X}_i$. We comment more on its history and examples in Section 3.1. Next we discuss related work. The most related results in the literature are developed in [55, 56]. The work [55] in particular develops a superlinearly convergent procedure for nonconvex feasibility problems, which solves a quadratic programming problem at each iteration. The algorithm is shown to converge superlinearly when the classical *transversality* property holds

$$\text{If } \sum_{i \in I} v_i = 0, \text{ for } v_i \in N_{\mathcal{X}_i}(\bar{x}), \text{ then } v_i = 0 \text{ for } i \in I, \tag{13}$$

and either of the following two conditions hold for $i \in I$:

1. the set $\mathcal{X}_i$ is a manifold;
2. the normal cone to $\mathcal{X}_i$ has a unique unit norm element near $\bar{x}$.

The first setting is most interesting. In this case, the following corollary holds.

Corollary 2.6. *Consider the setting of Corollary 2.5. Suppose that the set $\mathcal{X}_i$ is a C^2 manifold for all $i \in I$ and that the family intersects transversely at $\bar{x}$ in the sense of* (13)*. Then Items 1 and 2 (with $\eta = 1$) of Corollary 2.5 hold.*

Thus, in the case of transversal manifold intersections, Algorithm 4 converges superlinearly (in fact, quadratically) under the same setting as [55]. Beyond the manifold setting, Corollary 2.5 provides additional consequences for semialgebraic intersections. Finally, we mention one benefit of Algorithm 4 compared to the algorithm of [55]: each step solves a linear system, rather than a quadratic programming problem.

Outline of the rest of the paper. Having stated all of our main results, we now turn to proofs and a brief numerical study. First, Section 3 studies the (b)-regularity property, providing basic examples and proving calculus rules. Next, Section 4 proves all the algorithmic results stated in this section. Finally, Section 5 presents a brief numerical study and describes several implementation strategies.

3 The (b)-regularity property: examples and calculus

In this section, we present basic examples and calculus for the (b)-regularity property in Assumption (A2). This property was recently studied in the manuscript [15], focusing on functions and the Clarke subdifferential. In the following definition, we broaden the concept to mappings.[1]

Definition 3.1 ((b)-regularity along a set $\mathcal{Y}$)**.** Consider a locally Lipschitz mapping $F\colon \mathbb{R}^d \to \mathbb{R}^m$, a set $\mathcal{Y}$ and a nonempty-valued $G\colon \mathbb{R}^d \rightrightarrows \mathbb{R}^{m\times d}$. Fix a point $\bar{x} \in \mathcal{Y}$ and a scalar $\eta > 0$. Then the pair (F, G) is (b)*-regular along* $\mathcal{Y}$ *at* $\bar{x}$ *with exponent* $1+\eta$ if there exists $C > 0$ such that the estimate

$$\|F(x) + A(y - x) - F(y)\| \leq C\|x - y\|^{1+\eta} \tag{14}$$

holds for all x near $\bar{x}$, $A \in G(x)$, and $y \in \mathcal{Y}$ near $\bar{x}$.

3.1 Examples

A natural choice for the mapping G in Definition 3.1 is simply the Clarke Jacobian: $G = \partial F$. More generally, "generalized Jacobian mappings" can arise from automatic differentiation routines. Recently, Bolte and Pauwels [5] developed a mathematical model for such routines. In their work they identified that the output of such routines are often *conservative set-valued vector fields,* as formalized in the following definition.

Definition 3.2 (Conservative set-valued vector fields.)**.** Consider a locally Lipschitz mapping $F\colon \mathbb{R}^d \to \mathbb{R}^m$ and a set-valued mapping $G\colon \mathbb{R}^d \rightrightarrows \mathbb{R}^{m\times d}$ with nonempty compact-values and closed graph. Then G is a *conservative set-valued vector field for* F if for any absolutely continuous curve $x\colon [0, 1] \to \mathbb{R}^d$, we have

$$\frac{d}{dt}F(x(t)) = Ax(t) \qquad \text{for a.e. } t \in [0, 1] \text{ and all } A \in G(x(t)).$$

As shown by [5] (with precursors in [16, 21]), the Clarke Jacobian ∂F is a conservative set-valued vector field for any semialgebraic mapping F, though other examples are possible [45]. The later work [14] then showed that whenever both F and G are semialgebraic, conservative set-valued vector fields satisfy the (b)-regularity along singleton sets. This is quoted in the following lemma, consisting of several basic examples of Definition 3.1.

Lemma 3.1 (Basic Examples)**.** *Suppose that* $F\colon \mathbb{R}^d \to \mathbb{R}^m$ *is a locally Lipschitz mapping. Fix a point* $\bar{x} \in \mathbb{R}^d$ *and closed sets* $\mathcal{Y} \subseteq \mathbb{R}^d$ *containing* $\bar{x}$.

1. **(Smooth mappings)** *If* F *is* C^1 *near* $\bar{x}$ *and the Jacobian* ∇F *is locally Lipschitz, then the pair* $(F, \nabla F)$ *is (b)-regular along* $\mathcal{Y}$ *at* $\bar{x}$ *with exponent* 2.

[1] Note the slight discrepancy with Assumption (A2): in the terminology of this section, the pair (f, g^{T}) is (b)-regular along $\mathcal{X}_*$ at $\bar{x}$ with exponent $1+\eta$.

2. **(Sublinear functions)** *If $m = 1$, the mapping F is sublinear, and $\mathcal{Y} = \mathrm{Lin}(F)$, then for all $\eta > 0$ the pair $(F, \partial F)$ is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta$.*

3. **(Semialgebraic mappings)** *If F is semialgebraic, $\mathcal{Y} = \{\bar{x}\}$, and G is a semialgebraic conservative set-valued vector field for F (e.g., $G = \partial F$), then there exists $\eta > 0$ such that the pair (F, G) is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta$.*

Proof. Item 1 is straightforward so we omit the proof. Item 2 is the shown in [15, Lemma 2.6.1]. Item 3 is shown in [14] (the case of $G = \partial F$ is shown in [6]). □

Finally we give several examples involving distance functions. Here we present a key sufficient condition – Equation (15). This condition, which was first introduced in [73, 74] for manifolds and recently studied for general sets in [15], is the classical notion of (b)-regularity for two nested sets.

Lemma 3.2 (Distance Functions)**.** *Fix a point $\bar{x} \in \mathbb{R}^d$ and closed sets $\mathcal{Y} \subseteq \mathcal{X}$ in $\mathbb{R}^d$ containing $\bar{x}$. Consider the following condition: there exists $C, \eta > 0$ such that*

$$|\langle v, y - x\rangle| \leq C\|v\|\|y - x\|^{1+\eta}, \tag{15}$$

for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ near $\bar{x}$ and all $v \in N_{\mathcal{X}}(x)$. Define $G := \partial \mathrm{dist}(\cdot, \mathcal{X})^{\mathsf{T}}$. Then the pair $(\mathrm{dist}(\cdot, \mathcal{X}), G)$ is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta$ if and only if (15) holds. In particular, (15) holds automatically when

1. *$\mathcal{X}$ is semialgebraic and $\mathcal{Y} = \{\bar{x}\}$.*

2. *$\mathcal{X} = \mathcal{Y}$ and $\mathcal{X}$ is a C^2 manifold around $\bar{x}$ (with $\eta = 1$).*

3. *$\mathcal{X}$ is a convex cone and $\mathcal{Y} = \mathcal{X} \cap (-\mathcal{X})$ is its lineality space (with $\eta = 1$).*

Proof. Note that whenever the pair is (b)-regular, the estimate (15) trivially follows from (5). Now we prove that (15) implies the pair is (b)-regular. To that end, let $\delta > 0$ be small enough that the estimate (15) holds for $x \in B_\delta(\bar{x}) \cap \mathcal{X}$ and $y \in B_{2\delta}(\bar{x}) \cap \mathcal{Y}$. Let us first suppose that $z \in B_\delta(\bar{x}) \backslash \mathcal{X}$ and let

$$v_z := \frac{z - \hat{z}}{\|z - \hat{z}\|} \in \partial \mathrm{dist}(z, \mathcal{X}), \qquad \text{for some } \hat{z} \in P_{\mathcal{X}}(z).$$

Recall that $v_z \in N_{\mathcal{X}}(\hat{z})$. Thus, since $\hat{z} \in B_{2\delta}(\bar{x})$, we have

$$|\mathrm{dist}(z, \mathcal{X}) + \langle v_z, y - z\rangle| = |\mathrm{dist}(z, \mathcal{X}) + \langle v_z, \hat{z} - z\rangle| + |\langle v_z, y - \hat{z}\rangle| \leq C\|y - \hat{z}\|^{1+\eta}$$

since $\mathrm{dist}(z, \mathcal{X}) + \langle v_z, \hat{z} - z\rangle = 0$. Now observe that

$$\|y - \hat{z}\|^{1+\eta} \leq 2\|y - z\|^{1+\eta} + 2\|z - \hat{z}\|^{1+\eta} \leq 4\|y - z\|^{1+\eta}.$$

Consequently, (b)-regularity with subgradient v_z then follows from the bound:

$$|\mathrm{dist}(z, \mathcal{X}) + \langle v_z, y - z\rangle| \leq 4C\|y - z\|^{1+\eta}.$$

Since $\partial \text{dist}(z, \mathcal{X}) = \text{conv}\, \frac{z - P_{\mathcal{X}}(z)}{\text{dist}(z,\mathcal{X})}$, the (b)-regularity estimate with arbitrary $v \in \partial \text{dist}(z, \mathcal{X})$ follows from averaging the above bound over all possible projections $\hat{z}$. Next, let $z \in \mathcal{X} \cap B_\delta(\bar{x})$. Then $\partial \text{dist}(z, \mathcal{X}) = N_{\mathcal{X}}(z) \cap \bar{B}$. Thus, the (b)-regularity estimate is precisely the estimate (15).

We now prove the Items. First note that Item 1 follows from the work [6] applied to distance functions. Second, Item 3 follows from [15, Proposition 2.3.1]. Finally, we prove Item 2. Suppose that $\mathcal{X} = \mathcal{Y}$ and $\mathcal{X}$ is a C^2-smooth manifold around $\bar{x}$. Then there exists $C > 0$ such that $y - x \in T_{\mathcal{X}}(x) + C\|y - x\|^2 \bar{B}$ for all $x, y \in \mathcal{X}$ near $\bar{x}$. Thus, we have $|\langle v, y - x\rangle| \leq C\|v\|\|x - y\|^2$ for all $x, y \in \mathcal{X}$ near $\bar{x}$ and $v \in N_{\mathcal{X}}(x)$, as desired. □

3.2 Calculus

Next we turn our attention to a few basic calculus results. The following theorem develops a chain rule for (b)-regularity.

Theorem 3.3 (Chain rule)**.** *Consider two locally Lipschitz mappings $F_1 \colon \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and $F_2 \colon \mathbb{R}^{d_2} \to \mathbb{R}^{d_3}$, and define the composition $F = F_2 \circ F_1$. Consider two locally bounded set-valued mappings $G_1 \colon \mathbb{R}^{d_1} \rightrightarrows \mathbb{R}^{d_2 \times d_1}$ and $G_2 \colon \mathbb{R}^{d_2} \rightrightarrows \mathbb{R}^{d_3 \times d_2}$ and define the composition*

$$G(x) := \{A_2 A_1 \colon A_1 \in G_1(x) \text{ and } A_2 \in G_2(F_1(x))\} \qquad \text{for all } x \in \mathbb{R}^d.$$

Fix a set $\mathcal{Y}_2 \subset \mathbb{R}^{d_2}$, define $\mathcal{Y}_1 := F_1^{-1}(\mathcal{Y}_2)$, and let $\bar{x} \in \mathcal{Y}_1$. Suppose that

1. *(F_1, G_1) is (b)-regular along $\mathcal{Y}_1$ at $\bar{x}$ with exponent $1 + \eta_1$.*
2. *(F_2, G_2) is (b)-regular along $\mathcal{Y}_2$ at $F_1(\bar{x})$ with exponent $1 + \eta_2$.*

Then (F, G) is (b)-regular along $\mathcal{Y}_1$ at $\bar{x}$ with exponent $1 + \min\{\eta_1, \eta_2\}$.

Proof. Let U be a neighborhood of $F_1(\bar{x})$ and $C > 0$ be a constant such that

$$\|F_2(z') - (F_2(z) + A_2(z' - z))\| \leq C\|z' - z\|^{1+\eta_2}$$

for all $z \in U$, $A_2 \in G_2(z)$, and $z' \in U \cap \mathcal{Y}_2$. Let $V = F_1^{-1}(U)$ and let $V' \subseteq V$ be a neighborhood of $\bar{x}$ small enough that there exists $\beta > 0$ with

$$\|F_1(y) - (F_1(x) + A_1(y - x))\| \leq \beta\|x - y\|^{1+\eta_1}$$

for all $x \in V'$, $A_1 \in G_1(x)$, and $y \in V' \cap \mathcal{Y}_1$. Now, given $x \in V'$, select any $A_1 \in G_1(x)$ and $A_2 \in G_2(F_1(x))$. Let $L > 0$ satisfy $L \geq \sup_{A \in G_2(F_1(V'))} \|A\|_2$. In addition, assume that L is a Lipschitz constant for F_1 on V'. Then for all $x \in V'$ and $y \in V' \cap \mathcal{Y}_1$, we have

$$\begin{aligned}
&|F(y) - (F(x) + A_2 A_1(y - x))| \\
&\leq |F_2(F_1(y)) - (F_2(F_1(x)) + A_2(F_1(y) - F_1(x)))| + \|A_2\|_2 \|F_1(y) - (F_1(x) + A_1(y - x))\| \\
&\leq C\|F_1(y) - F_1(x)\|^{1+\eta_2} + \beta L\|x - y\|^{1+\eta_1} \\
&\leq CL^{1+\eta_2}\|y - x\|^{1+\eta_2} + \beta L\|x - y\|^{1+\eta_1},
\end{aligned}$$

where the third inequality follows from the inclusions $x \in V'$, $F_1(x) \in U$, $y \in V' \cap \mathcal{Y}_1$, and $F_1(y) \in U \cap \mathcal{Y}_2$. The proof then follows. □

The chain rule immediately leads to leads to a sum-rule. The proof is routine, so we omit it.

Corollary 3.3 (Sum Rule)**.** *Consider locally Lipschitz mappings $F_i \colon \mathbb{R}^d \to \mathbb{R}^m$, locally bounded set-valued mappings $G_i \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^{m\times d}$, and sets $\mathcal{Y}_i \subseteq \mathbb{R}^d$ indexed by a finite set I. Define the set $\mathcal{Y} := \bigcap_{i\in I} \mathcal{Y}_i$, the mapping $F = \sum_{i\in I} F_i$, and the mapping*

$$G(x) = \left\{ \sum_{i\in I} A_i \colon A_i \in G_i(x) \text{ for } i \in I \right\} \qquad \text{for all } x \in \mathbb{R}^d.$$

Suppose that for each $i \in I$, the pair (F_i, G_i) is (b)-regular along $\mathcal{Y}_i$ at $\bar{x}$ with exponent $1 + \eta_i$. Then (F, G) is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \min_{i\in I}\{\eta_i\}$.

The final result of this section states that (b)-regularity is preserved by "stacking" mappings. The proof is straightforward, so we omit it.

Lemma 3.4 (Stacking)**.** *Consider locally Lipschitz mappings $F_1 \colon \mathbb{R}^d \to \mathbb{R}^{d_1}$ and $F_2 \colon \mathbb{R}^d \to \mathbb{R}^{d_2}$ and define the mapping $F(x) := (F_1(x), F_2(x))$ for all $x \in \mathbb{R}^d$. Consider two locally bounded set-valued mappings $G_1 \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^{d_1\times d}$ and $G_2 \colon \mathbb{R}^d \rightrightarrows \mathbb{R}^{d_2\times d}$ and define the mapping*

$$G(x) = \left\{ \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} : A_1 \in G_1(x) \text{ and } A_2 \in G_2(x) \right\} \qquad \text{for all } x \in \mathbb{R}^d.$$

Fix sets $\mathcal{Y}_1 \subseteq \mathbb{R}^d$ and $\mathcal{Y}_2 \subseteq \mathbb{R}^d$ and define $\mathcal{Y} := \mathcal{Y}_1 \cap \mathcal{Y}_2$. Suppose that for $i \in \{1, 2\}$ the pair (F_i, G_i) is (b)-regular along $\mathcal{Y}_i$ at $\bar{x}$ with exponent $1 + \eta_i$. Then (F, G) is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \min\{\eta_1, \eta_2\}$.

To close this section, we mention that further calculus rules (e.g., preservation under spectral lifts) may be adapted from those in [15].

3.3 Consequences for semialgebraic mappings

Given the chain rule and the basic examples of Lemma 3.1, we have the following immediate consequences for semialgebraic mappings.

Corollary 3.5 (Chain-rule with semialgebraic mappings)**.** *Suppose that $F_1 \colon \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and $F_2 \colon \mathbb{R}^{d_2} \to \mathbb{R}^{d_3}$ are locally Lipschitz mappings. Let $\bar{x} \in \mathbb{R}^{d_1}$ and define the set $\mathcal{Y} := F_1^{-1}(F_1(\bar{x}))$. Suppose that*

1. *the mapping F_2 is semialgebraic and $G_2 \colon \mathbb{R}^{d_2} \rightrightarrows \mathbb{R}^{d_3\times d_2}$ is a semialgebraic conservative set-valued vector field for F_2 (e.g., ∂F_2).*

2. *and that either of the following hold:*

   (a) *near $\bar{x}$ the mapping F_1 is C^1 and the Jacobian $G_1 := \nabla F_1$ is Lipschitz.*

   (b) *there exists a locally bounded set-valued mapping $G_1 \colon \mathbb{R}^{d_1} \rightrightarrows \mathbb{R}^{d_2\times d_1}$ such that (F_1, G_1) is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta_1$.*

Define $G(x) := \{A_2A_1 \colon A_1 \in G_1(x)$ and $A_2 \in G_2(F_1(x))\}$ for all $x \in \mathbb{R}^d$. Then there exists $\eta > 0$ such that $(F_2 \circ F_1, G)$ is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta$.

As in Corollary 3.3, the result of Corollary 3.5 leads to a "sum rule" for semialgebraic mappings, whose details are immediate.

3.4 Consequences for functions

In this section, we prove the claims of Section 2.1.3.

3.4.1 Proof of Proposition 2.1.

The result follows from Corollary 3.5 with mappings $F_2 := h$ and $F_1 := F(x)$.

3.4.2 Proof of Proposition 2.2.

Define the mapping $F = F_2 \circ F_1$ and $G(x) := \{A_2 \nabla F_1(x) \colon \text{ and } A_2 \in \partial F_2(F_1(x))\}$. By Corollary 3.5, the pair (F, G) is (b)-regular along $\mathcal{Y}$ at $\bar{x}$ with exponent $1 + \eta$. Applying Corollary 3.5 again to the composition $\|F(x)\|$ gives the result.

3.4.3 Proof of Proposition 2.3.

Recall that $g_i(x) \in \partial \text{dist}(x, \mathcal{X}_i)$. Consequently, by Lemma 3.1 there exists $\eta_i > 0$ such that the semialgebraic mappings $(\text{dist}(x, \mathcal{X}_i), g_i^\top)$ are (b)-regular along $\{\bar{x}\}$ at $\bar{x}$ with exponent $1 + \eta_i$ for $i \in I$. Therefore, the result follows by Corollary 3.3, as desired.

4 Proofs of the main algorithmic results

Throughout this section, we assume that (A1) and (A2) are in force. We begin with a few lemmata that will reappear in several proofs. The first Lemma ensures we can use the (b)-regularity estimate with $y = P_{\mathcal{X}_*}(x)$. The proof is straightforward, so we omit it.

Lemma 4.1. *Let $\delta' > 0$ and let $y \in \mathcal{X}_*$. We have*

$$x \in B_{\delta'}(y) \text{ and } \hat{x} \in P_{\mathcal{X}_*}(x) \implies \hat{x} \in B_{2\delta'}(y).$$

The next property is fundamental to the convergence of the `PolyakSGM` and `PolyakBundle` procedures. It states that negative subgradients aim towards the set of minimizers.

Lemma 4.2 (Aiming)**.** *Fix a point x satisfying the bound $\|x - \bar{x}\| \leq \delta$, as well as $\text{dist}(x, \mathcal{X}_*) \leq (\frac{\mu}{2C_{(b)}})^{1/\eta}$. Then*

$$\langle v, x - \hat{x} \rangle \geq \frac{\mu}{2} \cdot \text{dist}(x, \mathcal{X}_*) \qquad \text{for all } v \in g(x) \text{ and } \hat{x} \in P_{\mathcal{X}_*}(x). \tag{16}$$

In particular, if $x \notin \mathcal{X}_*$, the bound $\|v\| \geq \mu/2$ holds for all $v \in g(x)$.

Proof. By (b)-regularity and sharp growth, we have

$$C_{(b)}\|x-\hat{x}\|^{1+\eta}+\langle v, x-\hat{x}\rangle \geq f(x)-f(\hat{x}) \geq \mu\,\|x-\hat{x}\|.$$

Rearranging gives (16). The lower bound $\|v\| \geq \mu/2$ is immediate. □

Finally we derive the main consequence of (A3) that is used in this work. The following lemma shows that when iterated, algorithmic mappings generate iterates with two desirable properties: they do not travel far from $\bar{x}$ and they linearly converge to $\mathcal{X}_*$.

Lemma 4.3. *Let $\mathcal{A}$ satisfy* (A3). *Fix a point $x \in \mathbb{R}^d$ satisfying*

$$\|x-\bar{x}\| < \frac{(1-\rho)\varphi_1}{2} \qquad \textit{and} \qquad \operatorname{dist}(x, \mathcal{X}_*) < \varphi_2.$$

Define $z_{-1} := x$ and for all $i \geq 0$, define $z_i := \mathcal{A}^{\circ i}(x)$. Then for all $i \geq 0$ and $\hat{z}_i \in P_{\mathcal{X}_*}(z_i)$, we have

$$\|z_i-\bar{x}\| < \varphi_1 \qquad \textit{and} \qquad \operatorname{dist}(z_i, \mathcal{X}_*) \leq \|z_i-\hat{z}_{i-1}\| \leq \rho^i \operatorname{dist}(z_0, \mathcal{X}_*).$$

Proof. Assume without loss of generality that $f^*=0$ and define $z_{-1}=z_0$. We show that the following holds for all $i \geq 0$:

$$\|z_i-\bar{x}\| \leq \|z_0-\bar{x}\|\left(1+(1+\rho)\sum_{j=0}^{i-1}\rho^j\right) < \varphi_1, \tag{17a}$$

$$\|z_i-\hat{z}_{i-1}\| \leq \rho^i\,\|z_0-\hat{z}_0\|\,. \tag{17b}$$

The base case follows trivially. Now, assume the bounds hold up to some index i. Then, (17b) ensures $\operatorname{dist}(z_i, \mathcal{X}_*) \leq \operatorname{dist}(z_0, \mathcal{X}_*) < \varphi_2$. Therefore, by (A3), we have

$$\|z_{i+1}-\hat{z}_i\| \leq \rho\,\|z_i-\hat{z}_i\| \leq \rho^{i+1}\,\|z_0-\hat{z}_0\|\,,$$

which proves (17b). Finally, we have

$$\begin{aligned}\|z_{i+1}-\bar{x}\| &\leq \|z_{i+1}-\hat{z}_i\|+\|z_i-\hat{z}_i\|+\|z_i-\bar{x}\| \\ &\leq (1+\rho)\,\|z_i-\hat{z}_i\|+\|z_0-\bar{x}\|\left(1+(1+\rho)\sum_{j=0}^{i-1}\rho^j\right) \\ &\leq (1+\rho)\rho^i\,\|z_0-\hat{z}_0\|+\|z_0-\bar{x}\|\left(1+(1+\rho)\sum_{j=0}^{i-1}\rho^j\right) \\ &\leq \|z_0-\bar{x}\|\left(1+(1+\rho)\sum_{j=0}^{i}\rho^j\right) < \varphi_1,\end{aligned}$$

where the penultimate inequality follows from the bound $\|z_0-\hat{z}_0\|$ and the last inequality follows from $\|z_0-\bar{x}\| \leq (1-\rho)\varphi_1/2$. This proves (17a) and completes the proof. □

4.1 Proof of Theorem 2.1

The following lemma proves Theorem 2.1.

Lemma 4.4 (One step improvement)**.** *Let L be an upper bound for the maximal norm element of $g(B_\delta(\bar{x}))$. Let $s\colon \mathbb{R}^d \to \mathbb{R}^d$ satisfy $s(x) \in g(x)$ for all $x \in \mathbb{R}^d$. Define the mapping*

$$\mathcal{A}(x) := \begin{cases} x - \frac{f(x)-f^*}{\|s(x)\|^2} s(x) & \text{if } s(x) \neq 0; \\ x & \text{otherwise.} \end{cases}$$

Then $\mathcal{A}$ satisfies (A3) *with*

$$\varphi_1 = \delta; \quad \varphi_2 = \left(\frac{\mu}{4C_{(b)}}\right)^{1/\eta}; \quad \rho = \sqrt{1 - \frac{\mu^2}{4L^2}}.$$

Consequently, Theorem 2.1 holds.

Proof. Assume without loss of generality that $f^* = 0$. Fix a point $x \notin \mathcal{X}$ satisfying the bounds $\|x - \bar{x}\| < \varphi_1$ and $\operatorname{dist}(x, \mathcal{X}_*) < \varphi_2$. Notice that Lemma 4.2 guarantees that $s(x) \neq 0$. Choose $\hat{x} \in P_{\mathcal{X}_*}(x)$ and observe that

$$\begin{aligned} \|\mathcal{A}(x) - \hat{x}\|^2 &= \|x - \hat{x}\|^2 + \frac{f(x)^2}{\|v\|^2} - 2\frac{f(x)}{\|v\|^2}\langle v, x - \hat{x}\rangle \\ &= \|x - \hat{x}\|^2 + \frac{f(x)}{\|v\|^2}\left(f(x) + \langle v, \hat{x} - x\rangle\right) - \frac{f(x)}{\|v\|^2}\langle v, x - \hat{x}\rangle \\ &\leq \|x - \bar{x}\|^2 + \frac{f(x)}{\|v\|^2}\left(C_{(b)}\|x - \hat{x}\|^{1+\eta} - \frac{\mu}{2}\|x - \hat{x}\|\right) \\ &\leq \|x - \hat{x}\|^2 - \frac{f(x)}{\|v\|^2} \cdot \frac{\mu}{4}\|x - \hat{x}\| \\ &\leq \|x - \hat{x}\|^2 \left(1 - \frac{\mu^2}{4\|v\|^2}\right). \end{aligned}$$

where the first inequality follows from (b)-regularity and Lemma 4.2, the second inequality follows from the assumed bound for $\operatorname{dist}(x, \mathcal{X}_*)$, and the last inequality follows from sharpness. Thus, $\mathcal{A}$ satisfies (A3). Consequently, by Lemma 4.3, Theorem 2.1 holds. □

4.2 Proof of Theorem 2.2

We first establish some assumptions and notation. Without loss of generality we assume $f^* = 0$. We let $\{y_i\}_i$ and $\{v_i\}_i$ denote the iterates and generalized gradients generated by Algorithm 2. We let

$$\hat{y}_i \in P_{\mathcal{X}_*}(y_i) \qquad \text{for } i = 0, \ldots, d$$

denote projections of the iterates onto $\mathcal{X}_*$. Note that whenever $y_i \in B_{\delta/2}(\bar{x})$, we have $\hat{y}_i \in B_\delta(\bar{x})$ (see Lemma 4.1).

We now turn to several technical Lemmas and Propositions. The first proposition is proved in Appendix A.2. It will help us ensure that `PolyakBundle` terminates after at most d iterations.

Proposition 4.1. *Fix $i \geq 1$ and suppose that $\|P_{\ker(A_j)} v_j\| \geq \alpha > 0$ for all $j \leq i$. Then the following holds:*

$$\operatorname{rank}(A_{i+1}) = i + 1 \quad \textit{and} \quad \sigma_{i+1}(A_{i+1}) \geq \min\left\{\|A_0\|_2, 1\right\}\left(\frac{\alpha}{L\sqrt{2}}\right)^i.$$

We proceed with several technical Lemmas, whose proofs appear inline. First we show the following decomposition of $y_i - \hat{y}_0$, which we use repeatedly in the below.

Lemma 4.5. *For all $i \geq 1$, the following identity holds:*

$$y_i - \hat{y}_0 = P_{\ker(A_i)}(y_0 - \hat{y}_0) - A_i^\dagger \left[f(y_j) + \langle v_j, \hat{y}_0 - y_j\rangle\right]_{j=0}^i.$$

Proof. Recall the projection formula $I - A_i^\dagger A_i = P_{\ker(A_i)}$. The claimed decomposition follows since

$$\begin{aligned} y_i - \hat{y}_0 &= P_{\ker(A_i)}(y_0 - \hat{y}_0) + A_i^\dagger A_i(y_0 - \hat{y}_0) - A_i^\dagger \left[f(y_j) + \langle v_j, y_0 - y_j\rangle\right]_{i=0}^i \\ &= P_{\ker(A_i)}(y_0 - \hat{y}_0) - A_i^\dagger \left[f(y_j) + \langle v_j, \hat{y}_0 - y_j\rangle\right]_{i=0}^i, \end{aligned}$$

as desired. □

The second lemma shows that the update y_i improves upon y_0 whenever the gap vector $y_i - \hat{y}_0$ has a large component in $\ker(A_i)$.

Lemma 4.6 (Distance reduction)**.** *Fix $\gamma \leq \mu/2L$. Suppose that for some $i \geq 1$, we have*

$$\|P_{\ker(A_i)^\perp}(y_i - \hat{y}_0)\| \leq \gamma \|y_i - \hat{y}_i\|.$$

Then we have the following bound

$$\|y_i - \hat{y}_0\| \leq \sqrt{\frac{1 - \frac{\mu^2}{4L^2}}{1 - \gamma^2}}\, \|y_0 - \hat{y}_0\| \leq \|y_0 - \hat{y}_0\|.$$

Proof. By definition, we have $v_0 \perp \ker(A_i)$. In addition, by Lemma 4.2, we have the bound $\left|\left\langle \frac{v_0}{\|v_0\|}, y_0 - \hat{y}_0\right\rangle\right| \geq \frac{\mu}{2L}\|y_0 - \hat{y}_0\|$. Taken together, these imply

$$\|P_{\ker(A_i)^\perp}(y_0 - \hat{y}_0)\|^2 \geq \frac{\mu^2}{4L^2}\|y_0 - \hat{y}_0\|^2. \tag{18}$$

Next, observe that by Lemma 4.5, we have $P_{\ker(A_i)}(y_i - \hat{y}_0) = P_{\ker(A_i)}(y_0 - \hat{y}_0)$. Consequently, we have

$$\|y_i - \hat{y}_0\|^2 = \|P_{\ker(A_i)}(y_0 - \hat{y}_0)\|^2 + \|P_{\ker(A_i)^\perp}(y_i - \hat{y}_0)\|^2$$

$$
\begin{aligned}
&= \|y_0 - \hat{y}_0\|^2 - \|P_{\ker(A_i)^\perp}(y_0 - \hat{y}_0)\|^2 + \|P_{\ker(A_i)^\perp}(y_i - \hat{y}_0)\|^2 \\
&\leq \|y_0 - \hat{y}_0\|^2 \left(1 - \frac{\mu^2}{4L^2}\right) + \|P_{\ker(A_i)^\perp}(y_i - \hat{y}_0)\|^2 \\
&\leq \|y_0 - \hat{y}_0\|^2 \left(1 - \frac{\mu^2}{4L^2}\right) + \gamma^2 \|y_i - \hat{y}_0\|^2,
\end{aligned}
$$

where the penultimate inequality follows from (18) and the last inequality follows from $\|P_{\ker(A_i)^\perp}(y_i - \hat{y}_0)\| \leq \gamma \|y_i - \hat{y}_i\|$ and the bound $\|y_i - \hat{y}_i\| \leq \|y_i - \hat{y}_0\|$. Rearranging, we arrive at the desired conclusion. □

The third and final lemma is the core of our argument. It shows that `PolyakBundle` increases the rank of A_i until it finds a vector y_i that superlinearly improves upon y_0.

Lemma 4.7 (Alternatives)**.** *Fix $\gamma \leq \mu/4L$ and suppose that for all $j \leq i$, we have*

$$
\begin{aligned}
\|y_j - \hat{y}_0\| &\leq \|y_0 - \hat{y}_0\|; \\
\|P_{\ker(A_j)^\perp}(y_j - \hat{y}_0)\| &\leq \gamma \|y_j - \hat{y}_j\|; \\
\langle v_j, P_{\ker(A_j)}(y_j - \hat{y}_j)\rangle &\geq \frac{\mu}{8} \|y_j - \hat{y}_j\|;
\end{aligned}
\tag{19}
$$

and the inclusions $y_0 \in B_{\frac{\delta}{4}}(\bar{x})$, $y_j \in B_{\frac{3\delta}{4}}(\bar{x})$. Then

1. $\operatorname{rank}(A_{i+1}) = i+1$ *and*

$$
\|A_{i+1}^\dagger\|_2 \leq \max\left\{1, \frac{2}{\mu}\right\} \left(\frac{8\sqrt{2}L}{\mu}\right)^i.
$$

2. *At least one of the following hold:*

   (a) **(Maintain progress)** *We have the inequalities*

$$
\begin{aligned}
\|y_{i+1} - \hat{y}_0\| &\leq \|y_0 - \hat{y}_0\|; \\
\|P_{\ker(A_{i+1})^\perp}(y_{i+1} - \hat{y}_0)\| &\leq \gamma \|y_{i+1} - \hat{y}_{i+1}\|; \\
\langle v_{i+1}, P_{\ker(A_{i+1})}(y_{i+1} - \hat{y}_{i+1})\rangle &\geq \frac{\mu}{8} \|y_{i+1} - \hat{y}_{i+1}\|;
\end{aligned}
\tag{20}
$$

   and the inclusion $y_{i+1} \in B_{\frac{3\delta}{4}}(\bar{x})$.

   (b) **(Superlinear improvement I)***: the next iterate satisfies*

$$
\begin{aligned}
&\operatorname{dist}(y_{i+1}, \mathcal{X}_*) \leq \frac{8L}{\mu} \left(4^{1+\eta} C_{(b)} \sqrt{i+1} \|A_{i+1}^\dagger\|_2\right) \|y_0 - \hat{y}_0\|^{1+\eta}; \\
&\|y_{i+1} - \hat{y}_0\| \leq \|y_0 - \hat{y}_0\|.
\end{aligned}
$$

(c) **(Superlinear improvement II)***: the next iterate satisfies*

$$\operatorname{dist}(y_{i+1}, \mathcal{X}_*) \leq \frac{C_{(b)}\sqrt{i+1}}{\gamma} \left\|A_{i+1}^{\dagger}\right\|_2 \|y_0 - \hat{y}_0\|^{1+\eta};$$
$$\|y_{i+1} - \hat{y}_0\| \leq \|y_0 - \hat{y}_0\| + C_{(b)}\sqrt{i+1} \left\|A_{i+1}^{\dagger}\right\|_2 \|y_0 - \hat{y}_0\|^{1+\eta}.$$

Proof. We first prove Item 1. Observe that y_0 satisfies the conditions of Lemma 4.2 by assumption, so $\langle v_0, y_0 - \hat{y}_0\rangle \geq \frac{\mu}{2}\|y_0 - \hat{y}_0\|$. Consequently, since $A_0 = [v_0^\mathsf{T}]$, we have $\|A_0\|_2 \geq \frac{\mu}{2}$. Furthermore, the third inequality of (19) implies that $\|P_{\ker(A_j)}(v_j)\| \geq \frac{\mu}{8}$ for all $j \leq i$. Thus, Proposition 4.1 yields

$$\operatorname{rank}(A_{i+1}) = i+1, \quad \text{and} \quad \sigma_{\min}(A_{i+1}) = \sigma_{i+1}(A_{i+1}) \geq \min\left\{\frac{\mu}{2}, 1\right\} \cdot \left(\frac{\mu}{8\sqrt{2}L}\right)^i.$$

The inequality follows after noticing that $\|A^\dagger\|_2 = 1/\sigma_{\min}(A)$.

For the rest of the proof, we perform a case-by-case analysis.

Case 1: Suppose $\|P_{\ker(A_{i+1})^\perp}(y_{i+1} - \hat{y}_0)\| \geq \gamma\|y_{i+1} - \hat{y}_{i+1}\|$. When this holds, we have

$$\begin{aligned}\gamma\|y_{i+1} - \hat{y}_{i+1}\| &\leq \|P_{\ker(A_{i+1})^\perp}(y_{i+1} - \hat{y}_0)\| \\ &= \left\|A_{i+1}^\dagger\left[f(y_j) + \langle v_j, \hat{y}_0 - y_j\rangle\right]_{j=0}^i\right\| \\ &\leq \left\|A_{i+1}^\dagger\right\|_2 \sqrt{\sum_{j=0}^i C_{(b)}^2 \|\hat{y}_0 - y_j\|^{2(1+\eta)}} \\ &\leq C_{(b)}\sqrt{i+1}\left\|A_{i+1}^\dagger\right\|_2 \|y_0 - \hat{y}_0\|^{1+\eta}\end{aligned}$$

where the first inequality follows by assumption, the second inequality follows by (b)-regularity, and the third inequality follows from the assumption that $\|y_j - \hat{y}_0\| \leq \|y_0 - \hat{y}_0\|$ for all $j \leq i$. In addition, using Lemma 4.5 and the above inequality, we find that

$$\begin{aligned}\|y_i - \hat{y}_0\| &\leq \|y_0 - \hat{y}_0\| + \left\|A_{i+1}^\dagger\left[f(y_j) + \langle v_j, \hat{y}_0 - y_j\rangle\right]_{j=0}^i\right\| \\ &\leq \|y_0 - \hat{y}_0\| + C_{(b)}\sqrt{i+1}\left\|A_{i+1}^\dagger\right\|_2 \|y_0 - \hat{y}_0\|^{1+\eta},\end{aligned}$$

as desired.

Case 2: Suppose $\|P_{\ker(A_{i+1})^\perp}(y_{i+1} - \hat{y}_0)\| \leq \gamma\|y_{i+1} - \hat{y}_{i+1}\|$. Under this condition, Lemma 4.6 ensures that

$$\|y_{i+1} - \hat{y}_0\| \leq \|y_0 - \hat{y}_0\|. \tag{21}$$

This proves the first two inequalities of Item 2a. To prove the inclusion $y_{i+1} \in B_{\frac{3\delta}{4}}(\bar{x})$, note that since $y_0 \in B_{\frac{\delta}{4}}(\bar{x})$ and $\|y_0 - \hat{y}_0\| \leq \|y_0 - \bar{x}\|$, it follows that

$$\|y_{i+1} - \bar{x}\| \leq \|y_{i+1} - \hat{y}_0\| + \|\hat{y}_0 - y_0\| + \|y_0 - \bar{x}\| \leq 2\|y_0 - \hat{y}_0\| + \|y_0 - \bar{x}\| < \frac{3\delta}{4},$$

as desired.

In the remainder of the proof, we show that either we obtain local superlinear improvement or the lower bound (20) holds. To that end, we first note that

$$\begin{aligned}\frac{\mu}{2}\|y_{i+1}-\hat{y}_{i+1}\| &\le \langle v_{i+1}, y_{i+1}-\hat{y}_{i+1}\rangle \\ &= \langle v_{i+1}, P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_{i+1})\rangle + \langle v_{i+1}, P_{\ker(A_{i+1})}(y_{i+1}-\hat{y}_{i+1})\rangle \\ &\le L\|P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_{i+1})\| + \langle v_{i+1}, P_{\ker(A_{i+1})}(y_{i+1}-\hat{y}_{i+1})\rangle, \qquad (22)\end{aligned}$$

where the first inequality follows from the Lemma 4.2 and the third inequality follows from Cauchy-Schwarz. We now upper bound the first term in the right-hand side of (22).

Claim 1. *The following bound holds:*

$$\|P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_{i+1})\| \le (\mu/4L)\|y_{i+1}-\hat{y}_{i+1}\| + 4^{1+\eta}C_{(b)}\sqrt{i+1}\left\|A_{i+1}^\dagger\right\|_2\|y_0-\hat{y}_0\|^{1+\eta}.$$

Proof. First note that

$$\begin{aligned}\|P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_{i+1})\| &\le \|P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_0)\| + \|P_{\ker(A_{i+1})^\perp}(\hat{y}_{i+1}-\hat{y}_0)\| \\ &\le \gamma\|y_{i+1}-\hat{y}_{i+1}\| + \left\|A_{i+1}^\dagger A_{i+1}(\hat{y}_{i+1}-\hat{y}_0)\right\|, \qquad (23)\end{aligned}$$

where the second inequality follows from the assumption of this case and the projection identity $P_{\ker(A_{i+1})^\perp} = A_{i+1}^\dagger A_{i+1}$. We now upper bound $\left\|A_{i+1}^\dagger A_{i+1}(\hat{y}_{i+1}-\hat{y}_0)\right\|$ in (23). Indeed, (b)-regularity yields

$$|f(y_j) + \langle v_j, \hat{y}-y_j\rangle| \le C_{(b)}\|\hat{y}-y_j\|^{1+\eta} \qquad \text{for all } \hat{y}\in B_{2\delta}(\bar{x}) \text{ and } j\le i.$$

Consequently, for all $j \le i$, we have

$$\begin{aligned}|\langle v_j, \hat{y}_{i+1}-\hat{y}_0\rangle| &= |f(y_j)+\langle v_j,\hat{y}_{i+1}-y_j\rangle - (f(y_j)+\langle v_j,\hat{y}_0-y_j\rangle)| \\ &\le C_{(b)}\left(\|\hat{y}_0-y_j\|^{1+\eta} + \|\hat{y}_{i+1}-y_j\|^{1+\eta}\right) \\ &\le C_{(b)}\left(\|y_j-\hat{y}_0\|^{1+\eta} + (\|\hat{y}_{i+1}-y_{i+1}\| + \|y_{i+1}-\hat{y}_0\| + \|y_j-\hat{y}_0\|)^{1+\eta}\right) \\ &\le C_{(b)}\left(\|y_0-\hat{y}_0\|^{1+\eta} + (2\|y_{i+1}-\hat{y}_0\| + \|y_0-\hat{y}_0\|)^{1+\eta}\right) \\ &\le C_{(b)}4^{1+\eta}\|y_0-\hat{y}_0\|^{1+\eta},\end{aligned}$$

where the third inequality follows from (19) and the fourth inequality follows from (21). Now, since $A_{i+1} = [v_1 \quad \dots \quad v_i]^\mathsf{T}$, it follows that

$$\left\|A_{i+1}^\dagger A_{i+1}(\hat{y}_{i+1}-\hat{y}_0)\right\| = \left\|A_{i+1}^\dagger\right\|_2\sqrt{\sum_{j=0}^{i}\langle v_j,\hat{y}_{i+1}-\hat{y}_0\rangle^2} \le 4^{1+\eta}C_{(b)}\sqrt{i+1}\left\|A_{i+1}^\dagger\right\|_2\|y_0-\hat{y}_0\|^{1+\eta},$$

Returning to (23), we thus arrive at the bound

$$\|P_{\ker(A_{i+1})^\perp}(y_{i+1}-\hat{y}_{i+1})\| \le \gamma\|y_{i+1}-\hat{y}_{i+1}\| + 4^{1+\eta}C_{(b)}\sqrt{i+1}\left\|A_{i+1}^\dagger\right\|_2\|y_0-\hat{y}_0\|^{1+\eta}.$$

Noting that $\gamma L \le \mu/4$ yields the result. □

Therefore, plugging the conclusion of the claim into (22), we obtain

$$\frac{\mu}{4}\|y_{i+1}-\hat{y}_{i+1}\| \leq C\|y_0-\hat{y}_0\|^{1+\eta} + \langle v_{i+1}, P_{\ker(A_{i+1})}(y_{i+1}-\hat{y}_{i+1})\rangle \tag{24}$$

for the constant $C := 4^{1+\eta}LC_{(b)}\sqrt{i+1}\left\|A_{i+1}^{\dagger}\right\|_2$. We now analyze (24) in two scenarios: First suppose that $C\|y_0-\hat{y}_0\|^{1+\eta} \leq \frac{\mu}{8}\|y_{i+1}-\hat{y}_{i+1}\|$. Then upper bounding (24) and rearranging yields

$$\frac{\mu}{8}\|y_{i+1}-\hat{y}_{i+1}\| \leq \langle v_{i+1}, P_{\ker(A_{i+1})}(y_{i+1}-\hat{y}_{i+1})\rangle,$$

which proves (20). Thus, the conclusion of Item 2a follows. Otherwise, the conclusion of Item 2b follows by

$$\|y_{i+1}-\hat{y}_{i+1}\| \leq \frac{8C}{\mu}\|y_0-\hat{y}_0\|^{1+\eta},$$

and equation (21). This completes the proof of the lemma. □

We now complete the proof of the theorem. Let y_i be the first iterate such that Item 2a in Lemma 4.7 does not hold (such an iterate must exist since the rank of A_i increases at each iteration). We first show that $\tilde{x}$ exists and $f(\tilde{x}) \leq f(y_i)$. Indeed, by Items 2b and 2c there exists a constant $B > 0$ such that

$$\begin{aligned}\|y_i-\hat{y}_i\| &\leq B\|y_0-\hat{y}_0\|^{1+\eta};\\ \|y_i-\hat{y}_0\| &\leq \|y_0-\hat{y}_0\| + B\|y_0-\hat{y}_0\|^{1+\eta}.\end{aligned} \tag{25}$$

We now define the constant $C_{\mathsf{s}} := \frac{BL}{\mu^{1+\eta}}$ and assume that

$$\operatorname{dist}(y_0,\mathcal{X}_*) \leq \min\left\{\left(\frac{\mu}{2C_{(b)}}\right)^{1/\eta}, \left(\frac{\mu^{1-\eta}}{LC_{\mathsf{s}}}\right)^{1/\eta}\right\}.$$

Then $B\operatorname{dist}^{\eta}(y_0,\mathcal{X}_*) \leq 1$ since

$$B\operatorname{dist}^{\eta}(y_0,\mathcal{M}) \leq \frac{\mu^{1-\eta}}{LC_{\mathsf{s}}}\cdot\frac{C_{\mathsf{s}}\mu^{1+\eta}}{L} = \frac{\mu^2}{L^2} \leq 1.$$

Therefore by (25), we have $\|y_i-\hat{y}_0\| \leq 2\|y_0-\hat{y}_0\|$. Consequently,

$$\|y_i-y_0\| \leq \|y_i-\hat{y}_0\| + \|y_0-\hat{y}_0\| \leq 3\|y_0-\hat{y}_0\| \leq \frac{3}{\mu}f(y_0) \leq \tau f(y_0).$$

Thus, $\tilde{x}$ exists and y_i satisfies $f(\tilde{x}) = \min_{y_j:\|y_j-y_0\|\leq\tau f(y_0)} f(y_j) \leq f(y_i)$.

Next we prove $f(\tilde{x}) \leq f(y_i) \leq C_{\mathsf{s}}f(x)^{1+\eta}$. To that end, note that $y_i \in B_{\delta}(\bar{x})$. Indeed, since $y_0 \in B_{\frac{\delta}{4}}(\bar{x})$, we have

$$\|y_i-\bar{x}\| \leq \|y_i-y_0\| + \|y_0-\bar{x}\| \leq 4\|y_0-\bar{x}\| < \delta,$$

where the second inequality follows from $\|y_i-y_0\| \leq 3\|y_0-\hat{y}_0\|$ and the trivial bound $\|y_0-\hat{y}_0\| \leq \|y_0-\bar{x}\|$. Therefore, taking into account the Lipschitz continuity and sharpness of f on $B_{\delta}(\bar{x})$, we find that

$$f(\tilde{x}) \leq f(y_i) \leq L\|y_i-\hat{y}_i\| \leq BL\operatorname{dist}^{1+\eta}(y_0,\mathcal{M}) \leq \frac{BL}{\mu^{1+\eta}}f(y_0)^{1+\eta} = C_{\mathsf{s}}f(x)^{1+\eta}.$$

This completes the proof.

4.3 Proof of Theorem 2.4

We assume that (A1), (A2), and (A3) are in force in this section. In the forthcoming proofs, we assume without loss of generality that $f^* = 0$. We first show that the iterates $\{x_k\}_k$ exist and stay in a neighborhood of $\bar{x}$, so that every call to `FallbackAlg` produces a linearly convergent set of iterates; in turn, this shows that each iteration of Algorithm 4 must terminate.

Lemma 4.8. *The iterates $\{x_k\}_k$ exist and for all $k \geq 0$, satisfy*

$$\begin{aligned} f(x_{k+1}) &\leq \frac{1}{2} f(x_k); \\ \|x_k - \bar{x}\| &\leq \frac{1-\rho}{2} \cdot \min\left\{\frac{\delta}{4}, \varphi_1\right\}; \\ \operatorname{dist}(x_k, \mathcal{X}_*) &\leq \varphi_2. \end{aligned}$$

Proof. We begin with some some notation. Define the following four constants

$$\omega := \frac{1+\rho}{\mu(1-\rho)}; \quad \delta_1 := \min\left\{\frac{\delta}{4}, \varphi_1\right\}; \quad \delta_2 := \frac{\varphi_2}{1 + \max\left\{2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3}\right\}};$$

and

$$c := \left[\left(1 + \max\left\{2\kappa \frac{1+\rho}{1-\rho}, \frac{4L}{3}\right\}\right)\left(\frac{2}{1-\rho}\right)\right]^{-1}.$$

In particular, we have $\|x_0 - \bar{x}\| \leq c\delta_1$. Let $z_{j,k}$ denote the j^{th} iterate of the call to $\texttt{FallbackAlg}(x_k, \mathcal{A}, \frac{1}{2} f(x_k))$. In addition, for all j and k, let $\hat{z}_{j,k} \in P_{\mathcal{X}_*}(z_{j,k})$ and $\hat{x}_k \in P_{\mathcal{X}_*}(x_k)$. Now we turn to the proof.

If the iterates exist, clearly the inequality $f(x_{k+1}) \leq \frac{1}{2} f(x_k)$ holds for all $k \geq 0$. Thus, we focus on the latter two bounds. In particular, define $x_{-1} := x_0$. Then we claim the following three bounds hold for all $k \geq 0$:

$$\|x_k - x_{k-1}\| \leq c_k f(x_0); \tag{26a}$$

$$\|x_k - \bar{x}\| \leq c\delta_1 + f(x_0) \sum_{j=0}^{k-1} c_j; \tag{26b}$$

$$\operatorname{dist}(x_k, \mathcal{X}_*) \leq \delta_2 + f(x_0) \sum_{j=0}^{k-1} c_j. \tag{26c}$$

where we define $c_k := \max\left\{\omega\left(\frac{1}{2}\right)^k, \left(\frac{3}{4}\right)^k\right\}$ for all $k \geq 0$. We prove the claim by induction.

The base case is satisfied by assumption. Now assume that (26a), (26b), and (26c) hold up to index k. We first prove (26a). To that end, first suppose that we have $x_{k+1} = \texttt{PolyakBundle}(x_k, (3/2)^k)$. Then

$$\|x_{k+1} - x_k\| \leq \left(\frac{3}{2}\right)^k f(x_k) \leq \left(\frac{3}{4}\right)^k f(x_0) \leq c_k f(x_0), \tag{27}$$

where the first inequality follows from the definition of Algorithm 2 and the second inequality follows from the inductive hypothesis. Thus, in this case, (26a) holds. On the other hand, suppose that $x_{k+1} \neq \texttt{PolyakBundle}(x_k, (3/2)^k)$. To show that the fallback method initialized at x_k must terminate, we first bound $\|x_k - \bar{x}\|$ and $\operatorname{dist}(x_k, \mathcal{X}_*)$. To that end, the inductive hypothesis ensures

$$\|x_k - \bar{x}\| \leq c\delta_1 + f(x_0) \max\left\{2\omega, \frac{4}{3}\right\} \leq c\delta_1 \left(1 + \max\left\{2\kappa \cdot \frac{1+\rho}{1-\rho}, \frac{4L}{3}\right\}\right) \leq \frac{1-\rho}{2}\delta_1, \tag{28}$$

where the second inequality follows from the Lipschitz continuity of f and the third inequality follows by definition of c. A similar argument yields

$$\operatorname{dist}(x_k, \mathcal{X}_*) \leq \delta_2 + f(x_0) \max\left\{\frac{2(1+\rho)}{\mu(1-\rho)}, \frac{4}{3}\right\} \leq \varphi_2. \tag{29}$$

Then by (28), (29) and Lemma 4.3, the fallback algorithm must terminate by some iteration $i \in \mathbb{N}$. In addition, $x_{k+1} = z_{i,k}$ and $x_k = z_{0,k}$. Consequently,

$$\|x_{k+1} - x_k\| = \|z_{i,k} - z_{0,k}\| \leq \sum_{j=0}^{i-1} \|z_{j+1,k} - z_{j,k}\| \leq (1+\rho) \sum_{j=0}^{i-1} \|z_{j,k} - \hat{z}_{j,k}\|$$

where the second inequality follows from the triangle inequality and the following bound $\|z_{j+1,k} - \hat{z}_{j,k}\| \leq \rho \|z_{j,k} - \hat{z}_{j,k}\|$. Next, by Lemma 4.3, we have $\|z_{j,k} - \hat{z}_{j,k}\| \leq \rho^j \|z_{0,k} - \hat{z}_{0,k}\|$ for all $j < i$. Therefore,

$$\|x_{k+1} - x_k\| \leq (1+\rho) \sum_{j=0}^{i-1} \rho^j \|z_{0,k} - \hat{z}_{0,k}\| \leq \frac{1+\rho}{1-\rho} \|z_{0,k} - \hat{z}_{0,k}\| \leq \frac{1+\rho}{\mu(1-\rho)} f(x_k) \leq c_k f(x_0),$$

where the fourth inequality follows from sharpness and the inclusion $z_{0,k} \in B_{\frac{\delta}{4}}(\bar{x})$. This proves (26a).

Next, we prove (26b). To that end, observe that

$$\|x_{k+1} - \bar{x}\| \leq \|x_k - \bar{x}\| + \|x_{k+1} - x_k\| \leq c\delta_1 + f(x_0) \sum_{j=0}^{k-1} c_j + c_k f(x_0) = c\delta_1 + f(x_0) \cdot \sum_{j=0}^{k} c_j,$$

where the second inequality follows by the inductive assumption and (26a). This proves (26b).

Finally we prove (26c). To that end, observe that

$$\begin{aligned} \operatorname{dist}(x_{k+1}, \mathcal{X}_*) \leq \|x_{k+1} - \hat{x}_k\| \leq \operatorname{dist}(x_k, \mathcal{X}_*) + \|x_{k+1} - x_k\| &\leq \delta_2 + f(x_0) \cdot \sum_{j=0}^{k-1} c_j + c_k f(x_0) \\ &\leq \delta_2 + f(x_0) \cdot \sum_{j=0}^{k} c_j, \end{aligned}$$

where the third inequality follows by the inductive assumption and (26a). This proves (26c) and completes the proof. □

An immediate corollary of Lemmas 4.3 and 4.8 is the following:

Corollary 4.9. *Every call to algorithm to* `FallbackAlg` *in Algorithm 4 will terminate after at most* $\left\lceil \frac{1}{1-\rho} \log(2\kappa) \right\rceil$ *evaluations of* $\mathcal{A}$*.*

Finally, we show that all bundle steps are successful when $k \geq K_1$.

Lemma 4.10. *For all* $i \geq 0$*, we have the following:*

1. $x_{K_1+i} = \texttt{PolyakBundle}(x_{K_1+i-1}, (3/2)^{K_1+i})$;
2. $f(x_{K_1+i}) - f^* \leq 2^{-(1+\eta)^i}$.

Proof. We begin with Item 1. To that end, we first show that for all $k \geq K_1$, the iterate x_k and scalar $\tau = (3/2)^k$ satisfy the assumptions of Theorem 2.2. In particular, the vector $\tilde{x}$ exists and achieves the superlinear improvement (9). Indeed, Lemma 4.8 shows that $\|x_k - \bar{x}\| \leq \frac{\delta}{4}$ for any k. Furthermore, by the sharp growth condition (A1) and the definition of K_1, we have

$$\begin{aligned}\operatorname{dist}(x_k, \mathcal{X}_*) \leq \frac{f(x_k)}{\mu} \leq 2^{-K_1}\frac{f(x_0)}{\mu} &\leq \frac{1}{f(x_0)\max\left\{\frac{(2C_{(b)})^{1/\eta}}{\mu^{1+1/\eta}}, \left(\frac{LC_{\mathsf{s}}}{\mu}\right)^{1/\eta}\right\}}\frac{f(x_0)}{\mu} \\ &= \min\left\{\left(\frac{\mu}{2C_{(b)}}\right)^{1/\eta}, \left(\frac{\mu^{1-\eta}}{LC_{\mathsf{s}}}\right)^{1/\eta}\right\}.\end{aligned}$$

Finally, notice that $\left(\frac{3}{2}\right)^k > \frac{3}{\mu}$ for all $k \geq K_1$. Consequently, all the conditions of Theorem 2.2 are satisfied and thus the point $\tilde{x}$ exists and satisfies

$$\begin{aligned}f(\tilde{x}) \leq C_{\mathsf{s}} f(x_k)^{1+\eta} \leq C_{\mathsf{s}}\left(2^{-K_1} f(x_0)\right)^{\eta} f(x_k) &\leq C_{\mathsf{s}}\left(2^{-\log(f(x_0)\,(2C_{\mathsf{s}})^{1/\eta})} f(x_0)\right)^{\eta} f(x_k) \\ &\leq \frac{1}{2} f(x_k),\end{aligned}$$

where the second inequality follows from Lemma 4.8. This completes the proof of Item 1.

We now prove Item 2. Define a sequence $\{a_k\}_k$ by $a_k := f(x_k)$ for all $k \geq 0$. From Lemma 4.8 and Item 1,

$$a_{k+1} \leq \begin{cases} \frac{1}{2} a_k, & \text{if } k < K_1; \\ C_{\mathsf{s}} a_k^{1+\eta}, & \text{otherwise.} \end{cases} \tag{30}$$

In particular, by definition of K_1, we have

$$a_{K_1} \leq 2^{-K_1} a_0 \leq \frac{1}{2(C_{\mathsf{s}})^{1/\eta}}.$$

Thus, unfolding (30) shows that for all $i \geq 0$, we have

$$a_{K_1+i} \leq C_{\mathsf{s}} a_{K_1+i-1}^{1+\eta} \leq (C_{\mathsf{s}})^{\sum_{j=0}^{i-1}(1+\eta)^j} a_{K_1}^{(1+\eta)^i} \leq \left((C_{\mathsf{s}})^{1/\eta} a_{K_1}\right)^{(1+\eta)^i} \leq \left(\frac{1}{2}\right)^{(1+\eta)^i}.$$

This completes the proof of Item 2. □

To finish the proof, we tabulate the total number of evaluations of g and $\mathcal{A}$. To that end, note that the first K_1 iterations each require at most d evaluations of g and $\left\lceil \frac{1}{1-\rho} \log(2\kappa) \right\rceil$ evaluations of $\mathcal{A}$ by the definition of `PolyakBundle` and Corollary 4.9, respectively. Each remaining step of the algorithm simply calls `PolyakBundle`, which requires d evaluations of g. Therefore, since $f(x_{K_1+i}) \leq \epsilon$ whenever

$$i \geq \frac{\log \log_2 \left(\frac{1}{\epsilon}\right)}{\log(1+\eta)},$$

the algorithm requires at most

$$dK_1 + d \left\lceil \frac{\log \log_2 \left(\frac{1}{\epsilon}\right)}{\log(1+\eta)} \right\rceil,$$

evaluations of g. In addition, since $\mathcal{A}$ is only called during the first K_1 iterations, we evaluate $\mathcal{A}$ at most $\left\lceil \frac{1}{1-\rho} \log(2\kappa) \right\rceil K_1$ times. This completes the proof.

4.4 Proof of Corollary 2.3

The result is an immediate corollary of Lemma 4.4 and the Theorem 2.4

4.5 Proof of Corollary 2.4

Let us first assume that $\mathcal{X}_*$ is isolated at $\bar{x}$ and F is semialgebraic. In this case, Item 2 follows from Lemma 3.1. Next we show that f and g satisfy Assumption (A2) (the other assumption is immediate). Indeed, this immediately follows from Corollary 3.5 since the norm is semialgebraic.

4.6 Proof of Corollary 2.5

In either case (i) or (ii), Item 3.2 follows from Lemma 3.2. Next we show that f and g satisfy Assumption (A2) (the other assumption is immediate). Indeed, by Lemma 3.2, each pair $(\mathrm{dist}(\cdot, \mathcal{X}_i), g_{\mathcal{X}_i}^{\mathsf{T}})$ is (b)-regular along $\mathcal{X}_i$ at $\bar{x}$ with exponent $1+\eta$. Consequently, by Corollary 3.3 the pair (f, g^{T}) is (b)-regular along $\mathcal{X}_*$ at $\bar{x}$ with exponent $1+\eta$, as desired.

4.7 Proof of Corollary 2.6

Item 1 is classical and shown for example in [46]. Item 2 follows from Lemma 3.2.

5 Numerical study and implementation strategies

In this section, we present implementation strategies for the `SuperPolyak` algorithm and a brief numerical illustration. We begin with several implementation strategies.

5.1 Implementation strategies

In this section, we discuss several strategies that we found to improve the numerical performance of `SuperPolyak`.

5.1.1 Early termination of `PolyakBundle`.

We suggest terminating `PolyakBundle` early, returning some iterate y_i whenever at least one of the following holds.

1. (**Rank deficiency**) Suppose that there exists $i \leq d$ such that $\operatorname{rank}(A_j) = j$ for all $j < i$, but $\operatorname{rank}(A_i) < i$. In this case, we suggest that `PolyakBundle` return
$$\tilde{x} = \operatorname*{argmin}_{y_j \,:\, j \leq i \text{ and } \|y_j - y_0\| \leq \tau f(y_0)} f(y_j).$$
2. (**Large distance traveled**) Suppose that there exists $i < d$ such that $\|y_j - y_0\| \leq \tau f(y_0)$ for all $j < i$, but $\|y_i - y_0\| > \tau f(y_0)$. In this case, we suggest that `PolyakBundle` return
$$\tilde{x} = \operatorname*{argmin}_{y_j \,:\, j \leq i-1 \text{ and } \|y_j - y_0\| \leq \tau f(y_0)} f(y_j).$$
3. (**Superlinear improvement**) Suppose that we have an estimate η_{est} of the (b)-regularity exponent η and that $f(y_0) - f^* < 1$. Suppose that we find an iterate y_i such that
$$f(y_i) - f^* \leq (f(y_0) - f^*)^{1+\eta_{\mathsf{est}}}. \tag{31}$$
In this case, we suggest that `PolyakBundle` return the first such iterate $\tilde{x} = y_i$.

It is possible to show that these strategies still result in superlinear improvement, but we do not pursue this result here.

5.1.2 Updating η_{est} in `SuperPolyak`

Item 3 in Section 5.1.1 may never be triggered if η_{est} is too large. We suggest the following simple update strategy. Suppose that (31) fails for all candidates y_i but $x_{k+1} =$ `PolyakBundle`$(x_k, (3/2)^k)$. Then we update

$$\eta_{\mathsf{est}} = \max\left(\eta_{\mathsf{lb}}, q \cdot \eta_{\mathsf{est}}\right), \quad \text{where} \quad \eta_{\mathsf{lb}} \geq 0,\ q \in (0,1).$$

In our implementation, we set $\eta_{\mathsf{est}} = 1$, $\eta_{\mathsf{lb}} = 0.1$, and $q = 0.9$. It is straightforward to show that the iterates x_k continue converge to superlinearly with this estimation strategy.

5.1.3 Less frequent calls to `PolyakBundle`

In early iterations of Algorithm 4, the `PolyakBundle` procedure may not succeed. To reduce wasted computation, one may

1. Replace the factor $(3/2)^k$ in line 2 with ω^k for some $\omega > 1$;
2. Replace the factor $1/2$ in line 6 with some $0 < \gamma < 1$.

In particular, it is straightforward to show that the iterates x_k continue to converge superlinearly whenever $\omega\gamma < 1$. These changes have separate effects. Reducing ω makes the early termination strategy in Item 2 of the strategies outlined in Section 5.1.1 more likely to be triggered, lowering the cost of early unsuccessful `PolyakBundle` steps. On the other hand, reducing γ results in less frequent calls to `PolyakBundle`.

5.1.4 Computing `PolyakBundle` in $O(d^3)$ operations

Each step of `PolyakBundle` requires the evaluation of $A_i^{\dagger}w$ for some $w \in \mathbb{R}^i$. When $i \leq d$, the pseudoinverse of an $i \times d$ matrix may be computed explicitly in $O(di^2)$ floating point operations (flops) [31]. Consequently, using this method, one may compute all matrices $A_i^{\dagger}$ (for $i = 1, \ldots, d$) with $O(d^4)$ flops. In this section, we point out that there exists a more efficient method for "updating" the pseudoinverse $A_i^{\dagger}$. Each update requires $O(d^2)$ flops, bringing the total cost down to $O(d^3)$ flops. The method is based on iteratively updating the QR decomposition of A_i^{T}, which allows efficient evaluation of $A_i^{\dagger}w$ for arbitrary $w \in \mathbb{R}^i$. We place the proof and description of the algorithm in Appendix A.1.1.

Proposition 5.1. *Consider the setting of Proposition 2.2. Then there exists a method to return the point $\tilde{x} =$ `PolyakBundle`(x, τ) using at most $O(d^3)$ flops (ignoring the cost of evaluations of f and g).*

5.2 Numerical illustration

We now briefly illustrate the numerical performance of `SuperPolyak` on several signal recovery applications, including low-rank matrix sensing, max-linear regression, phase retrieval, and compressed sensing. In each experiment, we run `SuperPolyak` with a natural first-order fallback method, which we also use a baseline method for comparison against. Our main finding is that `SuperPolyak` often improves – both in oracle complexity and time – on several first-order fallback methods, including the Polyak subgradient method, the method of alternating projections, and the classical fixed-point iteration. Our code is available at the following URL: `https://github.com/COR-OPT/SuperPolyak.jl`.

Implementation Details. Throughout we use the default scaling factors $\omega = \frac{3}{2}$ and $\gamma = \frac{1}{2}$ described in Section 5.1.3. We also use the algorithmic enhancements from Section 5.1. In each experiment, we fix a minimizer $\bar{x}$ of f and initialize both `SuperPolyak` and the fallback method at a uniform random point x satisfying $\|\bar{x} - x\|/\|\bar{x}\| = 1$; an exception is the basis pursuit experiment of Section 5.2.4, which is initialized at the zero vector. In each experiment, the problem data and initializer are chosen randomly; we found that the depicted behavior was similar across multiple runs of the algorithm, so we plot only one instance in each figure. With the exception of the experiment in Figure 1, all generalized gradients g were computed via the automatic differentiation library `ReverseDiff.jl`. The experiments in Figures 3, 5, 6, and 7 were performed on an Intel Core i7-7700 CPU desktop with 16GB of RAM running Manjaro Linux. The experiment in Figure 1 was performed on a shared Intel Xeon E5-2680 (v3) cluster with a 16GB RAM limit running Ubuntu Linux. We used Julia v1.6.1 in both environments.

5.2.1 Low-rank matrix-sensing and `PolyakSGM`

In this problem, we observe a measurement vector $\bar{y} \in \mathbb{R}^m$ satisfying

$$\bar{y} = \mathcal{A}(\bar{M}) + \bar{\xi},$$

where $\bar{M} \in \mathbb{R}^{d\times d}$ is a fixed rank r matrix, $\mathcal{A} : \mathbb{R}^{d\times d} \to \mathbb{R}^m$ is a linear operator and $\bar{\xi} \in \mathbb{R}^m$ is a "noise" vector. The goal of the low-rank matrix sensing problem is to recover $\bar{M}$. Recoverability depends on the operator $\mathcal{A}$. In this section, we consider linear operators $\mathcal{A}$ with rows $\mathcal{A}_i$, satisfying

$$\mathcal{A}_i(M) = \langle \ell_i, Mr_i\rangle \qquad \text{for some } \ell_i, r_i \in \mathbb{R}^d \text{ and all } M \in \mathbb{R}^{d\times d}.$$

The work [10] analyzes the following objective for this problem class:

$$f(U,V) := \frac{1}{m}\left\|\mathcal{A}(UV^{\mathsf{T}}) - y\right\|_1 \qquad \text{for all } U, V \in \mathbb{R}^{d\times r}, \tag{32}$$

In particular, [10] shows that f satisfies (A1) when (i) ℓ_i, r_i are i.i.d. standard Gaussian vectors, (ii) $m \gtrsim rd$, and (iii) at most a small constant fraction of entries of $\bar{\xi}$ are nonzero. Moreover, any solution $(\bar{U}, \bar{V})$ satisfies $\bar{M} = \bar{U}\bar{V}^{\mathsf{T}}$. Thus, Proposition 2.1 implies that f satisfies (A2).

We perform experiments with `SuperPolyak` using `PolyakSGM` as the fallback method in two different settings. In both settings, we set $\bar{\xi} = 0$, which leads to optimal value $f^* = 0$. Note that even in this setting, the nonsmooth ℓ_1 is penalty is preferable to ℓ_2 since it leads to better conditioning [10]. Note also that the total number of parameters we optimize over is $2dr$.

1. **(Varying dimensions/ranks)** In this setting, we choose $\bar{M} = \bar{U}\bar{V}^{\mathsf{T}}$ where $\bar{U}, \bar{V} \in \mathbb{R}^{d\times r}$ are uniform random $d \times r$ matrices with orthonormal columns. In addition, we choose ℓ_i, r_i to be i.i.d. standard Gaussian vectors. Figure 3 then compares `SuperPolyak` to `PolyakSGM` to for varying (d, r) and $m = 3rd$.

2. **(Effect of conditioning of $\bar{M}$)** In this setting, we choose $\bar{M} = \bar{U}\Lambda\bar{V}^{\mathsf{T}}$ where $\Lambda \in \mathbb{R}^{r\times r}$ is a diagonal matrix with condition number $\tilde{\kappa}$ and $\bar{U}, \bar{V} \in \mathbb{R}^{d\times r}$ are uniform random $d \times r$ matrices with orthonormal columns. We then compare `SuperPolyak` to `PolyakSGM` to for varying $\tilde{\kappa}$ under two measurement models:

(a) **(Gaussian measurements/small dimension)** Figure 4 chooses ℓ_i, r_i to be i.i.d. standard Gaussian vectors. Here, $2dr = 4000$.

(b) **(Hadamard measurements/medium dimension)** Figure 1 presents a similar, but larger scale comparison using measurements ℓ_i and r_i from a random Hadamard ensemble (which allows for faster matrix vector products); see [24, Section 6.3] for a formal description of the measurements. Here, $2dr = 131{,}072$.

In the experiments, `SuperPolyak` converges superlinearly and requires only a fraction of the oracle calls to g compared `PolyakSGM`. With the exception of a low-dimensional instance in Figure 4, the phenomenon persists when comparing CPU times.

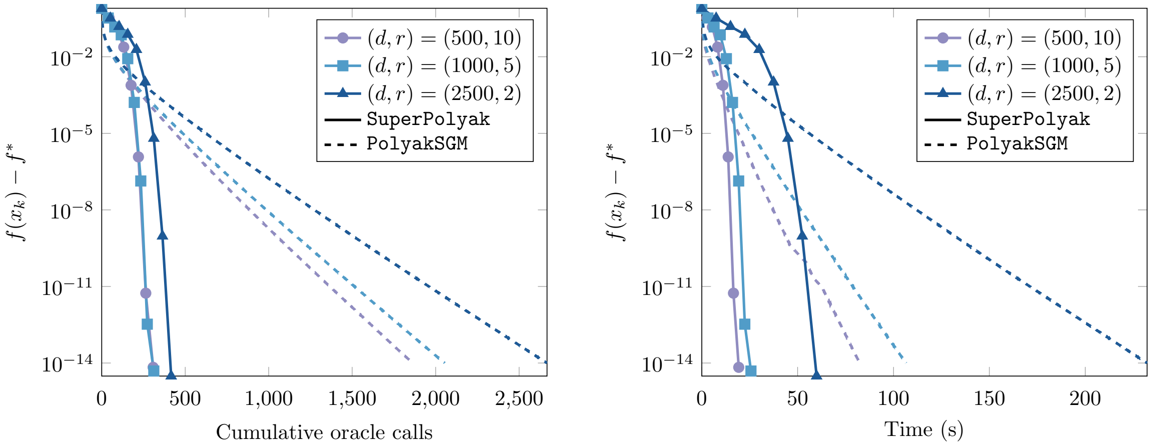

Figure 3: Low-rank matrix sensing with Gaussian measurements, varying dimension/ranks, and $m = 3rd$. See Section 5.2.1 for description.

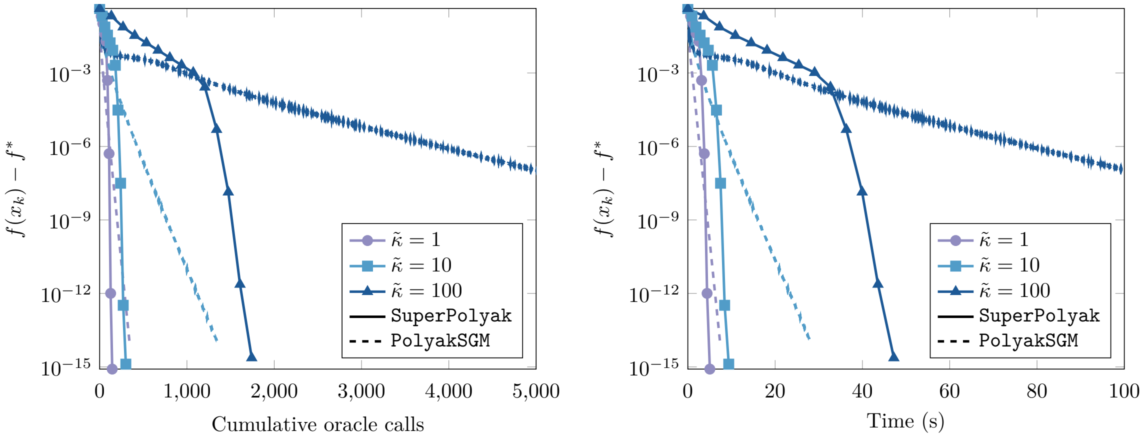

Figure 4: Low-rank matrix sensing with Gaussian measurements, varying condition number $\tilde{\kappa}$, and parameters $d = 500$, $r = 4$ and $m = 5rd$. See Section 5.2.1 for description.

5.2.2 Max-linear regression and `PolyakSGM`

In this problem, we observe a measurement vector $\bar{y} \in \mathbb{R}^m$ satisfying

$$\bar{y}_i = \max_{j \in [r]} \{\langle \bar{\beta}_j, a_i \rangle\} \qquad \text{for } i = 1, \ldots, m,$$

for known standard Gaussian vectors a_i $(i \in [m])$ and unknown vectors $\bar{\beta}_j$ $(j \in [r])$. This problem is an instance of the support function regression problem [32, 60], where we observe several random evaluations of the support function of $\operatorname{conv}\{\bar{\beta}_1, \ldots, \bar{\beta}_r\}$ and we seek to recover the vertices $\bar{\beta}_j$. To recover $\bar{\beta}_1, \ldots, \bar{\beta}_r$, we optimize the following objective

$$f(\beta_1, \ldots, \beta_r) := \frac{1}{m} \sum_{i=1}^{m} \left| y_i - \max_{j \in [r]} \{\langle \beta_j, a_i \rangle\} \right| \qquad \text{for all } \beta_1, \ldots, \beta_r \in \mathbb{R}^d.$$

We do not attempt to verify assumptions (A1) and (A2) for this problem class. Instead, we note that if the solution set is isolated, semialgebraicity of f implies that we (A2) holds (see Proposition 2.1). On the other hand, verifying Assumption (A1) is remains an intriguing open problem.

We now turn to our experiment. For simplicity, we sample each $\bar{\beta}_j$ uniformly from $\mathbb{S}^{d-1}$. Then in Figure 5, we apply `SuperPolyak` with fallback method `PolyakSGM`. Again we see that `SuperPolyak` outperforms the `PolyakSGM` method and appears insensitive to the number of problem parameters dr.

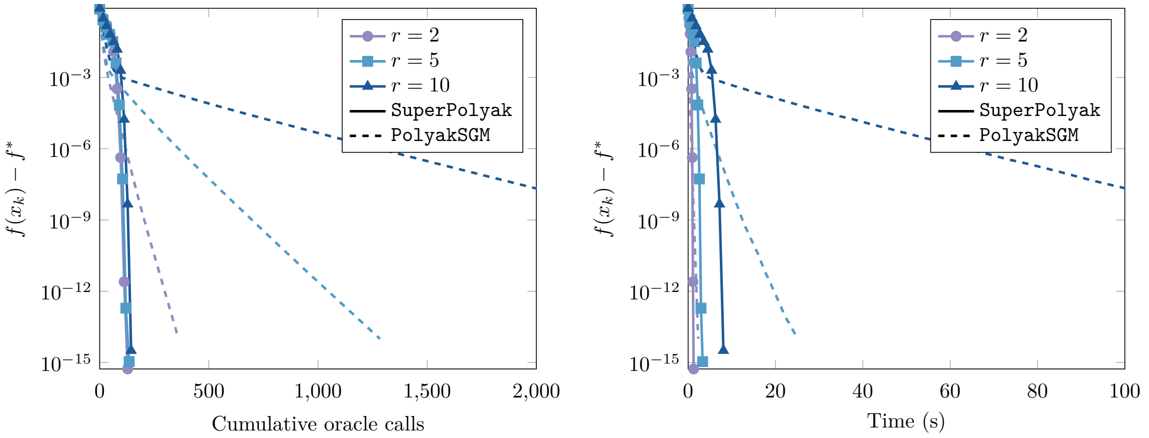

Figure 5: Max-linear regression with Gaussian measurements, varying r, and parameters $d = 500$ and $m = 3dr$. See Section 5.2.2 for description.

5.2.3 Phase retrieval and the method of alternating projections

In this problem, we observe a measurement vector $\bar{y} \in \mathbb{R}^m$ satisfying

$$\bar{y}_i = |\langle a_i, \bar{x} \rangle| \qquad \text{for } i = 1, \ldots, m,$$

for known measurement vectors a_i ($i \in [m]$) and an unknown signal $\bar{x} \in \mathbb{C}^d$, with the goal of recovering $\bar{x}$ up to phase.[2] To recover $\bar{x}$, we consider the feasibility formulation:

$$\text{find} \quad \hat{y} \in \mathcal{Y}_1 \cap \mathcal{Y}_2 \qquad \text{where} \qquad \mathcal{Y}_1 := \{u \in \mathbb{C}^m \mid |u| = y\}\,; \;\; \mathcal{Y}_2 := \operatorname{Range}(A); \tag{33}$$

and $A \in \mathbb{C}^{m \times d}$ is the matrix whose ith row is a_i^{H}. Given $\hat{y}$ in the intersection, we then estimate $\bar{x}$ with $\hat{x} = A^{\dagger}\hat{y}$. Note that when A is generic and $m \geq 4d - 4$, any such solution $\hat{x}$ is unique up to a global phase [2, 12]. To solve this feasibility formulation, we consider the following objective:

$$f(y) = \operatorname{dist}(y, \mathcal{Y}_1) + \operatorname{dist}(y, \mathcal{Y}_2).$$

Since $\mathcal{Y}_1$ and $\mathcal{Y}_2$ are smooth manifolds, Corollary 2.5 shows that f satisfies (A2) at any point $\hat{y} \in \mathcal{Y}_1 \cap \mathcal{Y}_2$. On the other hand, we were not able to locate property (A1) in the literature, even when the a_i follow a complex Gaussian distribution. Nevertheless, there is reason to believe it holds in the Gaussian setting, since the method of alternating projections (described in 2.2) locally linearly converges to an element of $\mathcal{Y}_1 \cap \mathcal{Y}_2$ [72].

We now turn to our experiment. We generate A with i.i.d. complex Gaussian entries and sample $\bar{x}$ uniformly from the unit sphere in $\mathbb{C}^d$, using $m = 4d$ measurements for varying dimension d. In Fig. 6, we apply `SuperPolyak` with the method of alternating projections (see Example 2.2) as the fallback method. Here, the oracle complexity of `SuperPolyak` is the number of evaluations of $P_{\mathcal{Y}_1} \circ P_{\mathcal{Y}_2}$ plus the number of subgradient evaluations of f. We see that `SuperPolyak` improves upon the method of alternating projections both in terms of oracle complexity and time.

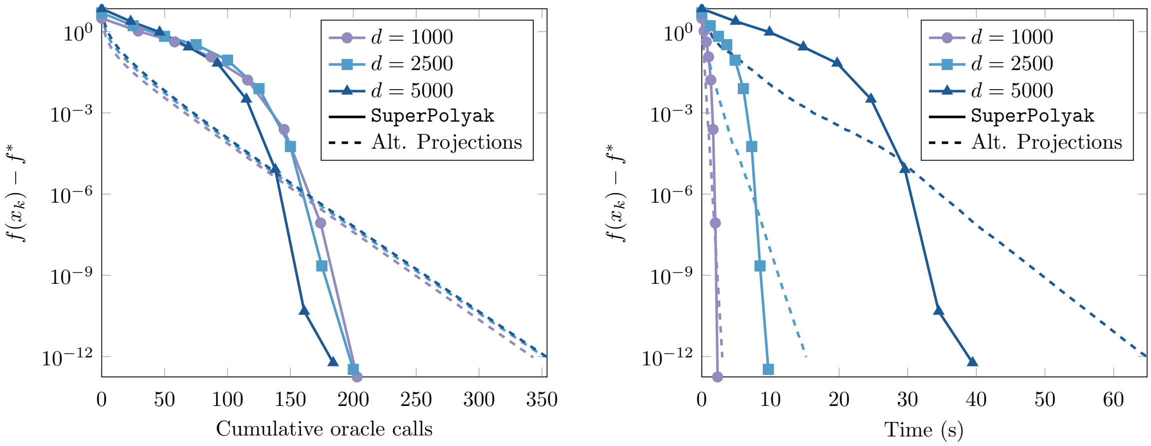

Figure 6: Complex phase retrieval with $m = 4d$ Gaussian measurements. See Section 5.2.3 for description.

[2] For this section, recall that $\langle x, y \rangle = \operatorname{Tr}(x^{\mathsf{H}}y)$ where x^{H} is the conjugate transpose of x and $|x| = \sqrt{\Re^2(x) + \Im^2(x)}$ for any x, $y \in \mathbb{C}^d$. As usual, we also identify $\mathbb{C}^d$ with $\mathbb{R}^{2d}$ in order to apply the results of this manuscript. To be consistent with the rest of the notation of the paper, we use i as an index, not the imaginary unit.

5.2.4 Compressed sensing and the proximal gradient method

In this problem, we observe a measurement vector $\bar{y} \in \mathbb{R}^m$ satisfying

$$\bar{y} = A\bar{x} + \xi,$$

where $A \in \mathbb{R}^{m \times d}$ is a known matrix, ξ is an unknown noise vector, $\bar{x}$ is an unknown sparse vector. The goal of the compressed sensing problem [19] is to recover $\bar{x}$ when m is on the order of the number of nonzeros of $\bar{x}$. There are several optimization based formulations for finding $\bar{x}$. For simplicity we focus on the "basis pursuit" formulation [11], which solves the following ℓ_1-penalized least squares problem:

$$h(x) := \frac{1}{2} \|Ax - y\|^2 + \lambda \|x\|_1 \qquad \text{for all } x \in \mathbb{R}^d.$$

A standard approach for minimizing h is the proximal gradient method, which iterates

$$x_{i+1} := T(x_i), \tag{34}$$

where for fixed $\tau > 0$ we define $T(x) := \operatorname{prox}_{\lambda\|\cdot\|_1}(x - \tau A^\mathsf{T}(Ax - y))$ for all $x \in \mathbb{R}^d$. This motivates us to consider the following objective

$$f(x) := \|x - Tx\| \qquad \text{for all } x \in \mathbb{R}^d,$$

which has the same minimizers as h and has minimal value $f^* = 0$. This objective satisfies (A1) automatically and the fixed-point iteration (34) is a valid algorithmic mapping in the sense of (A3); see [70]. Moreover, when A is drawn from a continuous distribution, the minimizer of f is unique for any positive λ with probability 1 [69]. Consequently, since f is semialgebraic, Proposition 2.2 shows that it satisfies (A2) (see also Corollary 3.5 below).

We now turn to our experiment. Here, we choose A with i.i.d. Gaussian entries and we vary dimension d, the number of nonzeros s of $\bar{x}$, and the number of measurements m. Then in Figure 7, we apply `SuperPolyak` with fallback method (34). Here, the oracle complexity of `SuperPolyak` is the number of evaluations of T plus the number of subgradient evaluations of f. We find that `SuperPolyak` converges superlinearly and outperforms the fixed-point iteration (34) in both oracle evaluations and CPU time.

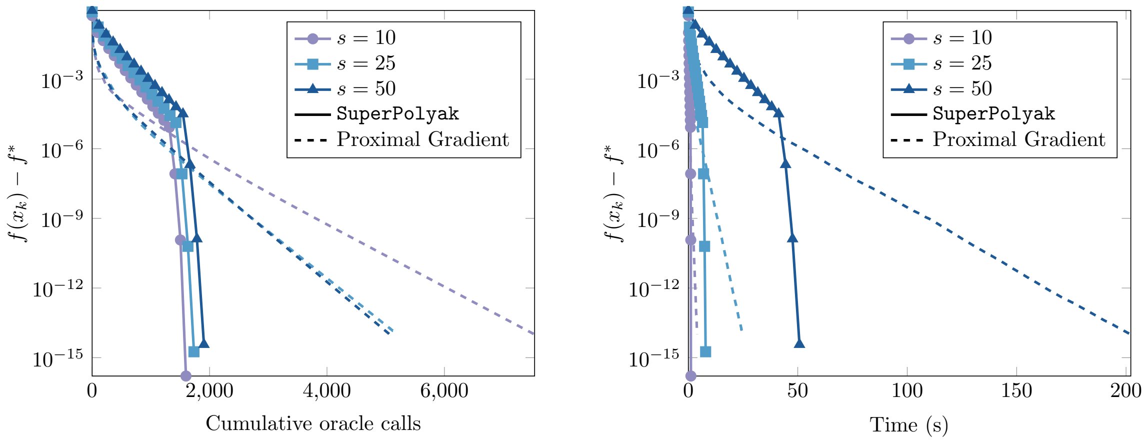

Figure 7: Basis pursuit with Gaussian measurements, varying sparsity s, and parameters $d = 100s$ and $m = 10s$. See Section 5.2.4 for description.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283, Savannah, GA, November 2016. USENIX Association.

[2] Radu Balan, Pete Casazza, and Dan Edidin. On signal reconstruction without phase. *Applied and Computational Harmonic Analysis*, 20(3):345–356, 2006.

[3] Stefan Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fund. math*, 3(1):133–181, 1922.

[4] Heinz H Bauschke, Dominikus Noll, and Hung M Phan. Linear and strong convergence of algorithms involving averaged nonexpansive operators. *Journal of Mathematical Analysis and Applications*, 421(1):1–20, 2015.

[5] Jérôme Bolte and Edouard Pauwels. Conservative set valued fields, automatic differentiation, stochastic gradient methods and deep learning. *Mathematical Programming*, 188(1):19–51, 2021.

[6] Jérôme Bolte, Aris Daniilidis, and Adrian Lewis. Tame functions are semismooth. *Mathematical Programming*, 117(1):5–19, 2009.

[7] Nicolas Boumal. An introduction to optimization on smooth manifolds. *Available online, Aug*, 2020.

[8] E.J. Candes and T. Tao. Near-optimal signal recovery from random projections: universal encoding strategies? *IEEE Trans. Inform. Theory*, 52(12):5406–5425, 2006.

[9] Emmanuel J Candes and Terence Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

[10] Vasileios Charisopoulos, Yudong Chen, Damek Davis, Mateo Díaz, Lijun Ding, and Dmitriy Drusvyatskiy. Low-rank matrix recovery with composite optimization: Good conditioning and rapid convergence. *Foundations of Computational Mathematics*, 2021.

[11] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[12] Aldo Conca, Dan Edidin, Milena Hering, and Cynthia Vinzant. An algebraic characterization of injectivity in phase retrieval. *Applied and Computational Harmonic Analysis*, 38(2):346–356, 2015.

[13] Chandler Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM J. Numer. Anal.*, 7(1):1–46, March 1970.

[14] Damek Davis and Dmitriy Drusvyatskiy. Conservative and semismooth derivatives are equivalent for semialgebraic maps. *Set-Valued and Variational Analysis*, 2021.

[15] Damek Davis, Dmitriy Drusvyatskiy, and Liwei Jiang. Subgradient methods near active manifolds: saddle point avoidance, local convergence, and asymptotic normality. *arXiv e-prints*, page arXiv:2108.11832, August 2021.

[16] Damek Davis, Dmitriy Drusvyatskiy, Sham Kakade, and Jason D Lee. Stochastic subgradient method converges on tame functions. *Foundations of computational mathematics*, 20(1):119–154, 2020.

[17] Damek Davis, Dmitriy Drusvyatskiy, Kellie J. MacPhee, and Courtney Paquette. Subgradient methods for sharp weakly convex functions. *Journal of Optimization Theory and Applications*, 179(3):962–982, 2018.

[18] Mateo Díaz and Benjamin Grimmer. Optimal convergence rates for the proximal bundle method. *arXiv preprint arXiv:2105.07874*, 2021.

[19] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[20] Dmitriy Drusvyatskiy. *Slope and geometry in variational mathematics.* PhD thesis, Cornell University, 2013.

[21] Dmitriy Drusvyatskiy, Alexander D Ioffe, and Adrian S Lewis. Curves of descent. *SIAM Journal on Control and Optimization*, 53(1):114–138, 2015.

[22] Dmitriy Drusvyatskiy, Alexander D Ioffe, and Adrian S Lewis. Transversality and alternating projections for nonconvex sets. *Foundations of Computational Mathematics*, 15(6):1637–1651, 2015.

[23] Yu Du and Andrzej Ruszczyński. Rate of convergence of the bundle method. *Journal of Optimization Theory and Applications*, 173(3):908–922, 2017.

[24] John C Duchi and Feng Ruan. Solving (most) of a set of quadratic equalities: composite optimization for robust phase retrieval. *Information and Inference: A Journal of the IMA*, 8(3):471–529, 09 2018.

[25] Grégory Emiel and Claudia Sagastizábal. Incremental-like bundle methods with application to energy planning. *Computational Optimization and Applications*, 46(2):305–332, 2010.

[26] I.I. Eremin. The relaxation method of solving systems of inequalities with convex functions on the left-hand side. *Dokl. Akad. Nauk SSSR*, 160:994–996, 1965.

[27] Yu M Ermol'ev and VI Norkin. Stochastic generalized gradient method for nonconvex nonsmooth stochastic optimization. *Cybernetics and Systems Analysis*, 34(2):196–215, 1998.

[28] Francisco Facchinei, Andreas Fischer, and Markus Herrich. An LP-Newton method: nonsmooth equations, KKT systems, and nonisolated solutions. *Mathematical Programming*, 146(1):1–36, 2014.

[29] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*. Springer Science & Business Media, 2007.

[30] J.L. Goffin. On convergence rates of subgradient optimization methods. *Math. Program.*, 13(3):329–347, 1977.

[31] Gene Golub and Charles Van Loan. Matrix computations, 2013.

[32] Adityanand Guntuboyina. Optimal rates of convergence for convex set estimation from support functions. *The Annals of Statistics*, 40(1):385–411, February 2012.

[33] Warren Hare and Claudia Sagastizábal. A redistributed proximal bundle method for nonconvex optimization. *SIAM Journal on Optimization*, 20(5):2442–2473, 2010.

[34] Alan J Hoffman. On approximate solutions of systems of linear inequalities. *Journal of Research of the National Bureau of Standards*, 49(4), 1952.

[35] Alexander D Ioffe. Variational analysis of regular mappings. *Springer Monographs in Mathematics. Springer, Cham*, 2017.

[36] Alexey F Izmailov and Mikhail V Solodov. *Newton-type methods for optimization and variational problems*. Springer, 2014.

[37] Patrick R. Johnstone and Pierre Moulin. Faster subgradient methods for functions with Hölderian growth. *Mathematical Programming*, 180(1):417–450, 2020.

[38] Krzysztof C Kiwiel. Efficiency of proximal bundle methods. *Journal of Optimization Theory and Applications*, 104(3):589–603, 2000.

[39] Krzysztof Czesław Kiwiel. A linearization algorithm for nonsmooth minimization. *Mathematics of Operations Research*, 10(2):185–194, 1985.

[40] Diethard Klatte and Bernd Kummer. *Nonsmooth equations in optimization: regularity, calculus, methods and applications*, volume 60. Springer Science & Business Media, 2006.

[41] Mark Aleksandrovich Krasnosel'skii. Two comments on the method of successive approximations. *Usp. Math. Nauk*, 10:123–127, 1955.

[42] Bernd Kummer. Newton's method for non-differentiable functions. *Mathematical research*, 45:114–125, 1988.

[43] John M Lee. Smooth manifolds. In *Introduction to Smooth Manifolds*, pages 1–31. Springer, 2013.

[44] Claude Lemarechal. An extension of davidon methods to non differentiable problems. In *Nondifferentiable optimization*, pages 95–109. Springer, 1975.

[45] Adrian Lewis and Tonghua Tian. The structure of conservative gradient fields. *arXiv preprint arXiv:2101.00699*, 2021.

[46] Adrian S. Lewis and Jérôme Malick. Alternating projections on manifolds. *Mathematics of Operations Research*, 33(1):216–234, 2008.

[47] Jiaming Liang and Renato DC Monteiro. A proximal bundle variant with optimal iteration-complexity for a large range of prox stepsizes. *SIAM Journal on Optimization*, 31(4):2955–2986, 2021.

[48] W Robert Mann. Mean value methods in iteration. *Proceedings of the American Mathematical Society*, 4(3):506–510, 1953.

[49] Robert Mifflin. Semismooth and semiconvex functions in constrained optimization. *SIAM J. Control Optim.*, 15(6):959–972, 1977.

[50] Robert Mifflin and Claudia Sagastizábal. A $\mathcal{VU}$-algorithm for convex minimization. *Mathematical Programming*, 104(2):583–608, 2005.

[51] V. I. Norkin. Stochastic generalized-differentiable functions in the problem of nonconvex nonsmooth stochastic optimization. *Cybernetics*, 22(6):804–809, 1986.

[52] E. A. Nurminskii. The quasigradient method for the solving of the nonlinear programming problems. *Cybernetics*, 9(1):145–150, Jan 1973.

[53] E. A. Nurminskii. Minimization of nondifferentiable functions in the presence of noise. *Cybernetics*, 10(4):619–621, Jul 1974.

[54] Welington de Oliveira and Claudia Sagastizábal. Bundle methods in the xxist century: A bird's-eye view. *Pesquisa Operacional*, 34(3):647–670, 2014.

[55] C. H. Jeffrey Pang. Nonconvex set intersection problems: From projection methods to the Newton method for super-regular sets. *arXiv e-prints*, page arXiv:1506.08246, 2015.

[56] CH Jeffrey Pang. Set intersection problems: supporting hyperplanes and quadratic programming. *Mathematical Programming*, 149(1):329–359, 2015.

[57] Jong-Shi Pang. Error bounds in mathematical programming. *Math. Program.*, 79(1–3):299–332, oct 1997.

[58] B. T. Polyak. Minimization of unsmooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9(3):14–29, 1969.

[59] Boris T. Polyak. Subgradient methods: a survey of Soviet research. In *Nonsmooth optimization (Proc. IIASA Workshop, Laxenburg, 1977)*, volume 3 of *IIASA Proc. Ser.*, pages 5–29. Pergamon, Oxford-New York, 1978.

[60] Jerry Ladd Prince and Alan S Willsky. Reconstructing convex sets from support line measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(4):377–389, 1990.

[61] Liqun Qi. Convergence analysis of some algorithms for solving nonsmooth equations. *Mathematics of operations research*, 18(1):227–244, 1993.

[62] Liqun Qi and Jie Sun. A nonsmooth version of Newton's method. *Mathematical programming*, 58(1):353–367, 1993.

[63] R.T. Rockafellar and R.J-B. Wets. *Variational Analysis*. Grundlehren der mathematischen Wissenschaften, Vol 317, Springer, Berlin, 1998.

[64] Claudia Sagastizábal. Divide to conquer: decomposition methods for energy optimization. *Mathematical programming*, 134(1):187–222, 2012.

[65] Robert Schreiber and Charles Van Loan. A storage-efficient WY representation for products of householder transformations. *SIAM J. Sci. and Stat. Comput.*, 10(1):53–57, 1989.

[66] N.Z. Shor. The rate of convergence of the method of the generalized gradient descent with expansion of space. *Kibernetika (Kiev)*, (2):80–85, 1970.

[67] S. Supittayapornpong and M.J. Neely. Staggered time average algorithm for stochastic non-smooth optimization with $O(1/t)$ convergence. *arXiv:1607.02842*, 2016.

[68] Andreas Themelis and Panagiotis Patrinos. Supermann: A superlinearly convergent algorithm for finding fixed points of nonexpansive operators. *IEEE Transactions on Automatic Control*, 64(12):4875–4890, 2019.

[69] Ryan J. Tibshirani. The lasso problem and uniqueness. *Electronic Journal of Statistics*, 7:1456 – 1490, 2013.

[70] Paul Tseng. Approximation accuracy, gradient methods, and error bound for structured convex optimization. *Mathematical Programming*, 125(2):263–295, 2010.

[71] J. von Neumann. *Functional Operators, Vol. II: The Geometry of Orthogonal Spaces.* Annals of Mathematics Studies, no. 22. Princeton University Press, Princeton, N. J., 1950.

[72] Irene Waldspurger. Phase retrieval with random gaussian sensing vectors by alternating projections. *IEEE Transactions on Information Theory*, 64(5):3301–3312, 2018.

[73] Hassler Whitney. Local properties of analytic varieties. In *Hassler Whitney Collected Papers*, pages 497–536. Springer, 1992.

[74] Hassler Whitney. Tangents to an analytic variety. In *Hassler Whitney Collected Papers*, pages 537–590. Springer, 1992.

[75] Philip Wolfe. A method of conjugate subgradients for minimizing nondifferentiable functions. In *Nondifferentiable optimization*, pages 145–173. Springer, 1975.

[76] Tianbao Yang and Qihang Lin. RSG: Beating subgradient method without smoothness and strong convexity. *Journal of Machine Learning Research*, 19(6):1–33, 2018.

A Proofs of auxiliary results

A.1 Proofs from Section 5

A.1.1 Proof of Proposition 5.1

In this section, we briefly sketch how to compute the iterates y_i of `PolyakBundle` by incrementally updating the "reduced QR decomposition" of $A^\intercal$. We begin with the following Lemma, which follows immediately from [31, Section 5.5.5].

Lemma A.1. *Consider $A \in \mathbb{R}^{m\times n}$ with $m \leq n$ and $\operatorname{rank}(A) = m$. Then $A^\dagger = QR^{-\intercal}$, where $A^\intercal = QR$ is the reduced QR decomposition of $A^\intercal$. Moreover, given any $b \in \mathbb{R}^m$, the vector $A^\dagger b$ can be computed in time $O(nm)$.*

From Lemma A.1, it follows that computing

$$y_i = y_0 - A_i^\dagger \left[f(y_j) - f^* + \langle v_j, y_0 - y_j\rangle\right]_{j=0}^i$$

in Algorithm 2 is possible in time $O(di)$ as long as the QR decomposition of A_i^T is available and A_i is full row rank. Recalling the Lemma 4.7, we observe that if A_i is rank deficient, we must have already obtained superlinear improvement. At that point, no further iterates y_i need be computed (as suggested in Section 5.1.1). Thus, we now sketch how to efficiently maintain the QR decomposition of A_i^T while A_i^T is full rank:

1. Initially, $A_1^\mathsf{T} = v_0$ and computing its QR factorization is trivial.
2. At step i, we have $A_{i+1}^\mathsf{T} = [A_i^\mathsf{T} \quad v_i]$. where A_i is full rank and its QR factorization is known. We consider the following cases:
   - If A_{i+1} remains full rank, we can compute its QR factorization with $O(d^2)$ flops using the algorithm from [31, Section 6.5.2].
   - If A_{i+1} becomes rank-deficient, we discard the maintained QR decomposition, compute the product $A_{i+1}^\dagger w$ explicitly using $O(d^3)$ flops, and exit the algorithm.

The above procedure requires $O(d^2)$ flops for every QR update step and $O(d^3)$ for applying $A_i^\dagger$ if A_i becomes rank-deficient. The former can happen at most d times, while the latter clearly happens at most once. Therefore, the total cost is $O(d^3)$ flops.

Finally, we briefly discuss the storage requirements of the above algorithm. The incremental update algorithm of [31, Section 6.5.2] requires computing the product $Q^\mathsf{T} v_i$, where Q is the $d \times d$ orthogonal matrix from the full QR factorization of A_i^T. Implemented naively, this requires storing $O(d^2)$ elements for Q. However, we can take advantage of the so-called *compact WYQ* format [65] to decompose Q as:

$$Q = I - UTU^\mathsf{T},$$

for certain $U \in \mathbb{R}^{d\times i}$ and upper triangular $T \in \mathbb{R}^{i\times i}$. Given U and T, we can compute $Q^\mathsf{T} v_i = v_i - U^\mathsf{T} T^\mathsf{T} U v_i$ in $O(di)$ flops; moreover, the compact WYQ representation can be updated in time $O(d^2)$ after adding a column to A_i^T. Therefore, the algorithm retains its computational complexity and requires storing at most $O(d\ell)$ numbers, where ℓ is the maximal iteration index.

A.2 Proofs from Section 4.2

A.2.1 Proof of Proposition 4.1

The proof is a consequence of the following lemma.

Lemma A.2. *Consider a matrix $A \in \mathbb{R}^{m\times d}$ and let $v \in \mathbb{R}^d$ satisfy*

$$\|v\| \le L; \quad \|P_{\ker(A)}(v)\| > \alpha > 0.$$

Suppose that $\operatorname{rank}(A) = k$ *for some* $k \leq d$*. Then the following holds:*

$$\sigma_{k+1}\left(\begin{bmatrix} A \\ v^\top \end{bmatrix}\right) \geq \frac{\alpha}{\sqrt{2}} \min\left\{1, \left(\frac{\sigma_k(A)}{L}\right)\right\}. \tag{35}$$

Proof. Define $w := \frac{P_{\ker(A)}v}{\|P_{\ker(A)}v\|}$ and $\bar{v} = \frac{v}{\|v\|}$. Observe that by the Davis-Kahan theorem [13]:

$$\|ww^\top - \bar{v}\bar{v}^\top\|_2 = \sqrt{1 - \langle w, \bar{v}\rangle^2} \leq \sqrt{1 - \frac{\alpha^2}{\|v\|^2}}. \tag{36}$$

Consequently, we have the following

$$\begin{aligned}
\sigma_{k+1}\left(\begin{bmatrix} A \\ v^\top \end{bmatrix}\right) &= \lambda_{k+1}(A^\top A + vv^\top) \\
&= \lambda_{k+1}(A^\top A + \|v\|^2 \bar{v}\bar{v}^\top) \\
&\geq \lambda_{k+1}(A^\top A + \min\{\sigma_k^2(A), \|v\|^2\} \bar{v}\bar{v}^\top) \\
&\geq \lambda_{k+1}(A^\top A + \min\{\sigma_k^2(A), \|v\|^2\} ww^\top) - \min\{\sigma_k^2(A), \|v\|^2\} \|ww^\top - \bar{v}\bar{v}^\top\|_2 \\
&= \min\{\sigma_k^2(A), \|v\|^2\} - \min\{\sigma_k^2(A), \|v\|^2\} \|ww^\top - \bar{v}\bar{v}^\top\|_2 \\
&\geq \min\{\sigma_k^2(A), \|v\|^2\} \left(1 - \sqrt{1 - \frac{\alpha^2}{\|v\|^2}}\right),
\end{aligned}$$

where the first inequality follows since eigenvalues preserve the Loewner order, the second inequality follows from Weyl's inequality, the third equality follows from the inclusion $w \in \ker(A)$ and $\operatorname{rank}(A) = k$, and the third inequality follows from (36). Finally, applying $\sqrt{1-x} \leq 1 - \frac{x}{2}$ to the lower bound above, we obtain

$$\lambda_{k+1}(A^\top A + vv^\top) \geq \frac{\min\{\sigma_k^2(A), \|v\|^2\} \alpha^2}{2\|v\|^2} \geq \min\left\{\frac{\sigma_k^2(A)\alpha^2}{2L}, \frac{\alpha^2}{2}\right\}$$

as desired. □

Proof of Proposition 4.1. The proof follows by iterating Lemma A.2 for all $i \leq k$. □