Convex optimization with p-norm oracles

Deeksha Adil DEADIL@ETHZ.CH
Institute for Theoretical Studies, ETH Zürich

Brian Bullins BBULLINS@PURDUE.EDU
Department of Computer Science, Purdue University

Arun Jambulapati JMBLPATI@GMAIL.COM
Independent Researcher

Aaron Sidford SIDFORD@STANFORD.EDU
Department of Computer Science, Stanford University

Editors: Matus Telgarsky and Jonathan Ullman

Abstract

In recent years, there have been significant advances in efficiently solving ℓ_s-regression using linear system solvers and ℓ_2-regression [Adil-Kyng-Peng-Sachdeva, J. ACM'24]. Would efficient smoothed ℓ_p-norm solvers lead to even faster rates for solving ℓ_s-regression when $2 \leq p < s$? In this paper, we give an affirmative answer to this question and show how to solve ℓ_s-regression using $\widetilde{O}(n^{\frac{\nu}{1+\nu}})$ iterations of solving smoothed ℓ_p regression problems, where $\nu := \frac{1}{p} - \frac{1}{s}$. To obtain this result, we provide improved accelerated rates for convex optimization problems when given access to an $\ell_p^s(\lambda)$-*proximal oracle*, which, for a point c, returns the solution of the regularized problem $\min_x f(x) + \lambda\|x - c\|_p^s$. Additionally, we show that these rates for the $\ell_p^s(\lambda)$-proximal oracle are optimal for algorithms that query in the span of the outputs of the oracle, and we further apply our techniques to settings of high-order and quasi-self-concordant optimization.

Keywords: Convex Optimization, Proximal Point Methods, Acceleration

1. Introduction

A prominent approach to efficiently solving convex optimization problems is to reduce them to solving a sequence of linear systems. For example, interior point methods reduce linear programming, semidefinite programming, many empirical risk minimization problems, and more to essentially solving a sequence of ℓ_2 *(or least-squares) regression problems* or, more broadly, to a sequence of linear systems (Karmarkar, 1984; Lee et al., 2019; Nesterov and Nemirovskii, 1994; Renegar, 1988; Wright, 1997). More recently, improvements in quasi-self-concordant (Bach, 2010; Carmon et al., 2020b; Karimireddy et al., 2018) and high-order (Bullins, 2020; Nesterov and Polyak, 2006; Nesterov, 2021) optimization have yielded new ways of using linear system solvers for solving well-established machine learning problems, including ℓ_∞ and logistic regression.

Similarly, recent advances in *iterative refinement* have shown how to reduce ℓ_p-regression, a natural data analysis problem, to solving a sequence of linear systems (Adil et al., 2019, 2024). ℓ_p regression problems appear in a wide range of applications, including semi-supervised learning (Alamgir and Luxburg, 2011; Flores et al., 2022; Kyng et al., 2015), data clustering and learning (Elmoataz et al., 2015, 2017), and low-rank matrix approximations (Chierichetti et al., 2017).

In this paper, we study a natural generalization: rather than reducing to ℓ_2-regression, i.e.,

$$\min_{x \in \mathbb{R}^d} \|Ax - b\|_2^2,$$

we consider reductions to *smoothed* ℓ_p-regression, i.e., problems of the form

$$\min_{x \in \mathbb{R}^d} g^\top x + \|Ax - b\|_2^2 + \|Cx - d\|_p^p .$$

We ask: *would smoothed ℓ_p-regression solvers yield faster rates for solving optimization problems?*

Addressing this question could advance the foundations of optimization theory and provide new tools for structured and high-order optimization beyond ℓ_p-regression. For example, as we will see, this question has connections to highly-smooth optimization in different norms, ball-acceleration in different geometries, and quasi-self-concordant optimization.

Efficient reductions from ℓ_s- to ℓ_p-regression. We consider the particular problem of efficiently reducing ℓ_s-regression to smoothed ℓ_p-regression for $2 \leq p < s$. We provide a new efficient reduction that solves this problem. Moreover, we obtain this result by an optimization theoretic framework of broader utility, which indeed has implications for structured optimization and higher-order optimization. We formally define ℓ_s regression in Definition 1 and then provide Theorem 2, which contains our main result for this problem (proved in Section 4).

Definition 1 (ℓ_s-Regression) *Given $2 \leq s < \infty$, $\epsilon > 0$, $A \in \mathbb{R}^{n \times d}$, and $b \in \mathbb{R}^n$, the* ℓ_s-regression problem *is to find $\widetilde{x} \in \mathbb{R}^d$ such that*

$$\|A\widetilde{x} - b\|_s^s \leq (1 + \epsilon) \min_{x \in \mathbb{R}^d} \|Ax - b\|_s^s. \tag{1}$$

We call such an $\tilde{x}$ an ϵ-approximate solution to the problem.

Theorem 2 (From ℓ_s-Regression to Smoothed ℓ_p-Regression) *There is an algorithm that, given $\epsilon > 0$, $s > p \geq 2$, computes an ϵ-approximate solution to ℓ_s-regression (Definition 1) in at most $O(s \cdot n^{\frac{\nu}{1+\nu}} \log^2 \frac{n}{\epsilon})$ iterations for $\nu := \frac{1}{p} - \frac{1}{s}$, each of which can be implemented in $O(d)$ time plus the time needed to solve $\tilde{O}(1)$[1] smoothed ℓ_p-regression problems of the form*

$$\min_{x \in \mathbb{R}^d} g^\top x + \|\tilde{A}x - \tilde{b}\|_2^2 + \|Cx - d\|_p^p .$$

Prior reductions include one of Bubeck et al. (2018) which established that $\tilde{O}(n^{\frac{1}{2} - \frac{1}{p}})$ iterations of comparable cost sufficed for $p = 2$. This was then improved by Adil et al. (2019) to $\tilde{O}(n^{\frac{p-2}{3p-2}})$, i.e., by a factor of $\Omega(n^{\frac{(p-2)^2}{2p(3p-2)}})$. For a general p, the previous state-of-the-art for such a reduction is due to Adil and Sachdeva (2020). Though not explicitly shown, their results imply an analogous Theorem 2 with an iteration bound of $\tilde{O}(n^{\frac{s\nu}{s-1}})$. Our work improves upon this rate for all $2 \leq p < s$ by a factor of $\Omega(n^{\frac{\nu^2}{1+\nu}})$ (similar to the prior improvement of Adil et al. (2019) over Bubeck et al. (2018) for $p = 2$).

1. We use $\tilde{O}$ to hide p, s, and $\mathrm{poly}\log n$ factors.

Optimal acceleration of $\ell_p^s(\lambda)$-proximal oracles. To prove Theorem 2, we consider a general optimization problem of independent interest. Specifically, we consider the problem of minimizing a convex function f given (approximate) access to what we call an $\ell_p^s(\lambda)$-*proximal oracle*.

Definition 3 ($\ell_p^s(\lambda)$-Proximal Oracle) *For $p, s \geq 2$, $\lambda > 0$, an* $\ell_p^s(\lambda)$-proximal oracle *for $f : \mathbb{R}^d \to \mathbb{R}$ is an oracle that, when queried at a "centering point" $c \in \mathbb{R}^d$, returns*

$$\tilde{x} \in \operatorname{argmin}_{x \in \mathbb{R}^d} f(x) + \lambda \|x - c\|_p^s.$$

This problem is well-studied in the special case when $p = s = 2$, where it corresponds to the standard (quadratic) proximal oracle (Parikh and Boyd, 2014). In particular, algorithms that access these oracles have been studied in convex optimization theory (see e.g., (Boyd et al., 2011; Combettes and Pesquet, 2007; Rockafellar, 1976)). A notable use of this oracle is in the *accelerated proximal point method* (Güler, 1992; Nesterov, 1983), which computes an ϵ-optimal solution with $O(\lambda^{1/2}\|x_0 - x^*\|_2 \epsilon^{-1/2})$ queries to the oracle, where $x^* \in \mathbb{R}^d$ is used to denote the minimizer of f throughout the paper. This query complexity is optimal (Nemirovskij and Yudin, 1983), and the method has played an important role in numerous algorithmic advances (Frostig et al., 2015; Lin et al., 2015; Parikh and Boyd, 2014; Schmidt et al., 2017; Shalev-Shwartz and Zhang, 2014), as well as in the high-order acceleration frameworks of Monteiro and Svaiter (2013); Gasnikov et al. (2019); Carmon et al. (2022), and the ball acceleration framework of Carmon et al. (2020b) (which has multiple applications (Carmon et al., 2021b, 2023, 2024; Jambulapati et al., 2024)).

However, outside of the above examples, optimal rates for solving convex optimization with $\ell_p^s(\lambda)$-proximal oracles are not known, to the best of our knowledge. This is particularly relevant for ℓ_p regression as, building upon an approach of Adil et al. (2024), solving ℓ_s-regression using the smoothed ℓ_p oracle reduces to acceleration with our $\ell_p^s(\lambda)$-proximal oracle. Consequently, to prove Theorem 2 we give new efficient algorithms (Algorithm 2), for which we prove the following.

Theorem 4 (Accelerated Optimization with $\ell_p^s(\lambda)$-Proximal Oracle) *Algorithm 2 given $s > p \geq 2$, $\epsilon > 0$, $x_0 \in \mathbb{R}^d$, outputs $\tilde{x} \in \mathbb{R}^d$ with $f(\tilde{x}) - f(x^\star) \leq \epsilon$ using $O((s\lambda\|x_0 - x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}})$ queries to an $\ell_p^s(\lambda)$-proximal oracle. Each iteration can be implemented in $O(d)$ time plus time of the $\ell_p^s(\lambda)$-proximal oracle.*

Though the rates of Theorem 4 may look unnatural at first glance, we in fact prove that, for any $p, s \geq 2$, they are optimal for broad classes of algorithms! Specifically, we show (Section 5) that there exists a function such that any zero-respecting algorithm, i.e., one that at every iteration queries an $\ell_p^s(\lambda)$-proximal oracle centered at a point $c \in \mathbb{R}^d$ which lies in the span of the outputs of the oracle, requires at least $\Omega((\lambda\|x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}})$ iterations to return an ϵ-suboptimal point.

Note that there is a broader literature of lower bounds for related problems, e.g., convex optimization in non-Euclidean norms with gradient oracles (Guzmán and Nemirovski, 2015) and in parallel settings (Diakonikolas and Guzmán, 2019). However, we are are unaware of such lower bounds for our particular $\ell_p^s(\lambda)$-proximal oracle.

Theorem 5 ($\ell_p^s(\lambda)$-Proximal Oracle Optimization Lower Bound) *For every $\ell_p^s(\lambda)$-proximal zero-respecting algorithm (Definition 22) given $p, s \geq 2$, $\lambda > 0$, $\epsilon > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\lambda\|x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}})$ iterations of the algorithm satisfies $f(x) - f(x^\star) \geq \epsilon$.*

Optimal acceleration of non-Euclidean ball-constrained optimization oracles. We show how our approach can also be used to show lower bounds for what we call ℓ_p *ball-constrained oracles*.

Definition 6 (ℓ_p Ball-constrained Oracle) *For $p \geq 2$, $r > 0$, an $\ell_p^\infty(r)$-proximal oracle (also referred to as an ℓ_p* ball-constrained oracle*) for $f : \mathbb{R}^d \to \mathbb{R}$ is an oracle that, when queried at a centering point $c \in \mathbb{R}^d$, returns*

$$\tilde{x} \in \operatorname{argmin}_{x \in \mathbb{R}^d | \|x-c\|_p \leq r} f(x).$$

We extend previous ℓ_2 ball oracle results (Carmon et al., 2020b, 2022; Karimireddy et al., 2018) to the ℓ_p setting (Section 3), and we also show that our rates are optimal for zero-respecting algorithms (Section 5).

Theorem 7 (Accelerated Optimization with ℓ_p Ball-constrained Oracle) *Algorithm 2 given $p \geq 2$, $r > 0$, $\epsilon > 0$, $x_0 \in \mathbb{R}^d$, outputs $\tilde{x}$ such that $f(\tilde{x}) - f(x^\star) \leq \epsilon$ using*

$$O((p\|x_0 - x^\star\|_p / r)^{\frac{p}{p+1}} \log(\|x_0 - x^\star\|_p^p / \epsilon))$$

queries to an $\ell_p^\infty(r)$-proximal oracle. Each iteration can be implemented in $O(d)$ time plus time of the $\ell_p^\infty(r)$-proximal oracle.

Theorem 8 (ℓ_p Ball-constrained Oracle Optimization Lower Bound) *For every $\ell_p^\infty(r)$-proximal zero-respecting algorithm (Definition 26) given $p \geq 2$, $r > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\|x^\star\|_p / r)^{\frac{p}{p+1}})$ iterations of the algorithm satisfies*

$$f(x) - f(x^\star) \geq \Omega(\|x^\star\|_p^{1/(p+1)} r^{p/(p+1)}) .$$

As previously mentioned, algorithms based on ℓ_2 ball-constrained oracles have had a wide range of applications. For example, they have arisen in solving natural problems in learning theory and machine learning, including logistic regression (Karimireddy et al., 2018; Carmon et al., 2020b), minimizing the maximum of a set of convex losses (Carmon et al., 2021b, 2024), and parallel stochastic optimization (Bubeck et al., 2019a; Carmon et al., 2023). Consequently, these results could lead to opportunities for improving these and related problems in different geometries for, e.g., high-order and quasi-self-concordant optimization settings. We further discuss in Section 4 how our methods can apply to such cases by showing that the oracles for these settings implement a more general oracle class.

Extensions to high-order optimization. As another application, we provide the following result, which extends our techniques to *highly smooth* problems with respect to the p-norm, i.e., problems whose q^{th}-order derivatives are L_q-Lipschitz continuous (for $q \geq 1$), in which case each step is based on minimizing a $\|\cdot\|_p^{q+1}$-regularized q^{th}-order Taylor expansion.[2]

Theorem 9 (High-order Optimization) *For $s > p \geq 2$, $q = s - 1$, let f be q^{th}-order L_q smooth with respect to $\|\cdot\|_p$, $x_0 \in \mathbb{R}^d$, and $\epsilon > 0$. Algorithm 2, implementing an $O(1)$-GMS$_p$ oracle by a (p, s)-Taylor oracle, finds $\tilde{x}$ such that $f(\tilde{x}) - f(x^\star) \leq \epsilon$ in*

$$O_{p,s}\left(\|x_0 - x^*\|_p^{\frac{1}{p(1+\nu)}} (L_q/\epsilon)^{\frac{1}{s(1+\nu)}} \right)$$

iterations. The cost of each iteration is at most $O(d)$ plus the cost of the (p, s)-Taylor oracle.

2. We use the notation $O_{p,s}(\cdot)$ to hide multiplicative running time factors which depend only on p and s.

We then show how this behaves, in a sense, as a certain *approximation* to the $\ell_p^s(\lambda)$-proximal oracle (for an appropriate choice of λ, which depends on the smoothness parameter L_q, and where $s = q + 1$), and the rates we obtain naturally generalize those found in previous results (Gasnikov et al., 2019; Monteiro and Svaiter, 2013). Such Euclidean-based high-order algorithms have been used, as previously discussed, to improve convergence for fundamental machine learning problems (Agarwal et al., 2017; Bullins, 2020; Nesterov and Polyak, 2006), and consequently rates for the non-Euclidean variants may have additional applications.

Simultaneous Independent Related Work. Independently, Contreras et al. (2025) also investigate non-Euclidean high-order and proximal point methods for convex optimization and provide similar algorithms and convergence rates. Specifically, they give algorithms for q^{th}-order (L, ν)-Hölder continuous functions for all $p, q \geq 1$, whereas our work focuses on the case where $q + 1 \geq p \geq 2$ for proximal oracles, ball-constrained optimization oracles, and highly smooth functions.Contreras et al. (2025) further establish lower bounds for minimizing q-th order (L, ν)-Hölder continuous functions with respect to $|| \cdot ||_p$, for algorithms that can query a local oracle, and as a result, their lower bounds depend on L. In contrast, our lower bounds are for (zero-respecting) ℓ_p^s proximal algorithms, and they place no smoothness restrictions on the function class. The initial draft of this paper was produced independently without having seen Contreras et al. (2025). Attempts were made to preserve independence during the revision process.

Future work and open problems. This paper provides improved rates for multiple convex optimization problems by using non-Euclidean optimization. However, these new rates come with costs. Whereas ℓ_2-regression can be solved by matrix multiplication and linear system solving, smoothed ℓ_p-regression is a more general and potentially complex optimization problem. Consequently, implementing steps of our algorithm efficiently and using this to obtain end-to-end runtime improvements for structured optimization problems is an interesting direction for future research.

Additionally, though we settle the complexity of convex optimization with the $\ell_p^s(\lambda)$-proximal oracle, this does not necessarily imply that our reduction from ℓ_s to smoothed ℓ_p-regression is optimal. Consider, for example, solving ℓ_6-regression using ℓ_2-regression and compare it with solving ℓ_6-regression via ℓ_4-regression. Our results state that the former would use $\approx n^{1/4}$ iterations of solving least squares regression, while the iteration complexity of the latter would improve to $\approx n^{1/13}$ (improving upon $\approx n^{1/10}$ iteration complexity due to Adil and Sachdeva (2020)), each of which would involve solving an ℓ_4-regression problem. Curiously, if we further solve each ℓ_4-regression problem using ℓ_2-regression, we would require $n^{1/5}$ such problems at each step, leading to a total of $n^{\frac{1}{13}+\frac{1}{5}} \geq n^{\frac{1}{4}}$. Improving our reductions to smoothed ℓ_p-regression, finding a similar type of reduction that doesn't occur this type of loss in repeated reduction, or providing more compelling evidence of optimality are additional interesting directions future research.

Furthermore, in the future, high-order optimization methods (e.g. (Gasnikov et al., 2019)) could potentially benefit from these tools, as such methods often depend on solving ℓ_p-norm regularized subproblems at each iteration. While there has been a line of work showing how subproblems based on second- (Nesterov and Polyak, 2006) and third-order (Nesterov, 2021) Taylor models admit more computationally efficient approximate solvers, less is known about more general settings. This work could potentially help in such settings.

Finally, obtaining similar improvements as the ones we have shown in this work for linear programming—or convex optimization with interior point methods more broadly—are additional exciting directions for future research. We hope our results help facilitate in these directions and,

with further research, may yield new non-Euclidean perspectives on basic problems and algorithms in machine learning, including those related to parallel, stochastic, and higher-order optimization, and ultimately lead to further efficiency gains in theory and in practice.

Paper overview. In Section 2 as a warm-up we give our main $\ell_p^s(\lambda)$-proximal point algorithm which includes a line-search in each iteration. In Section 3 we show how we may remove the need for this line-search and give an algorithm with a more general oracle. In Section 4 we use these algorithms to give rates for solving ℓ_s-regression and generalized high-order smooth problems. In Section 5, we give lower bounds that match the rates of our $\ell_p^s(\lambda)$-proximal point algorithms analyzed in Section 2.

2. Warm-up: Conceptual Proximal Point with Line Search

In this section, we analyze a simpler version of our generalized proximal point algorithm. Our aim is first to provide clarity for the convergence rates, while in the following section we address the issue of overall computational cost. Specifically, in this section we present an algorithm where each iteration involves finding points that satisfy certain implicit conditions, similar to previous works based on Monteiro-Svaiter acceleration (Monteiro and Svaiter, 2013). These implicit problems can be solved using an additional line-search procedure as in previous works (e.g., (Monteiro and Svaiter, 2013; Gasnikov et al., 2019; Carmon et al., 2020b)), followed by invoking an $\ell_p^s(\lambda)$-proximal oracle. We later show, in Section 3, how to modify the algorithm to obtain a line search-free method, and in doing so we introduce a more general oracle model. We further split our algorithm into two separate cases: $s < \infty$ and $s = \infty$ (the latter coinciding with an ℓ_p ball-constrained oracle).

Throughout the paper, we let $\|\cdot\|_p$ (for $p \in [1, \infty)$) denote the standard ℓ_p norm, and we use ω_p to denote, for an initial point $x_0 \in \mathbb{R}^d$, the Bregman divergence of the ℓ_p-norm function $\|x - x_0\|_p^p$, i.e., for any $x, y \in \mathbb{R}^d$,

$$\omega_p(x, y) = \|x - x_0\|_p^p - \|y - x_0\|_p^p - \langle \nabla_y \|y - x_0\|_p^p, x - y \rangle.$$

We also let $p^* := \frac{p}{p-1}$ throughout the paper.

We will prove the guarantees of Algorithm 1, which finds x such that $f(x) - f(x^*) \leq \epsilon$ for a convex function f via an $\ell_p^s(\lambda)$-proximal oracle given a line-search to estimate λ_t's. Our algorithm, in every iteration maintains two points x_t and z_t, takes a carefully chosen convex combination of the points to get y_t, applies the $\ell_p^s(\lambda)$-proximal oracle at y_t to compute x_{t+1}, and then uses the gradient of f at x_{t+1} to compute z_{t+1}. The main guarantee of the algorithm is given below in Theorem 10.

Theorem 10 *Let $s < \infty$. Given f, $x_0 \in \mathbb{R}^d$, and $\epsilon > 0$, there exist, for all $t \geq 0$, λ_t, x_{t+1} satisfying the conditions of Algorithm 1, and it outputs x_T such that $f(x_T) - f(x^\star) \leq \epsilon$, in*

$$O\left(\frac{s^{s(1+\nu)}}{p^{s\nu}} \cdot \frac{\lambda \|x_0 - x^*\|_p^s}{\epsilon}\right)^{\frac{1}{s(1+\nu)}} \quad \textit{iterations.}$$

When $s = \infty$, i.e., the algorithm is equipped with an $\ell_p^\infty(r)$-proximal oracle, we prove the following.

Algorithm 1 Conceptual Proximal Point Algorithm

Initialize: $a_0 = 1$, $A_0 = 0$, $z_0 = y_0 = x_0$, $T \geq 1$

for $t = 0, \cdots, T-1$ **do**

Find $\lambda_t > 0$, $x_{t+1} \in \mathbb{R}^d$ for which the following hold:

- $5^{p-1}\lambda_t a_{t+1}^p = A_{t+1}^{p-1}$, $A_{t+1} = A_t + a_{t+1}$, and $y_t = \frac{A_t}{A_{t+1}} x_t + \frac{a_{t+1}}{A_{t+1}} z_t$
- $\lambda_t = \begin{cases} \lambda \|x_{t+1} - y_t\|_p^{s-p} & \text{if } s < \infty \\ \|\nabla f(x_{t+1})\|_{\frac{p}{p-1}} / r^{p-1} & \text{if } s = \infty \end{cases}$ and $x_{t+1} \leftarrow \begin{cases} \arg\min_y f(y) + \frac{\lambda}{p}\|y - y_t\|_p^s & \text{if } s < \infty \\ \arg\min_{\|y - y_t\|_p \leq r} f(y) & \text{if } s = \infty \end{cases}$

$z_{t+1} \leftarrow \arg\min_{z \in \mathbb{R}^d} \sum_{i \in [t+1]} a_i \nabla f(x_i)^\top (z - y_i) + \|z - y_0\|_p^p$

end

return x_T

Theorem 11 *Let $s = \infty$. Given $f, x_0 \in \mathbb{R}^d$ and $\epsilon > 0$, there exist, for all $t \geq 0$, λ_t, x_{t+1} satisfying the conditions of Algorithm 1, and it outputs x_T such that $f(x_T) - f(x^\star) \leq \epsilon$, in*

$$O\left(\left(\frac{p\|x^\star - x_0\|_p}{r}\right)^{\frac{p}{p+1}} \log\left(\frac{\|x^\star - x_0\|_p}{\epsilon}\right)\right) \text{ iterations.}$$

The crux of the proofs of Theorem 10 and Theorem 11 is a standard potential argument (Diakonikolas and Orecchia, 2019; Nesterov et al., 2018). Namely, we rely on the following lemma, which shows how the potential $A_t(f(x_t) - f(x^\star)) + \omega_p(x^\star, z_t)$ decreases in every iteration of the algorithm. The proof can be found in Appendix B.

Lemma 12 *For all $t \geq 0$, the iterates of Algorithm 1 satisfy*

$$A_{t+1}(f(x_{t+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \leq A_t(f(x_t) - f(x^\star)) + \omega_p(x^\star, z_t) - A_{t+1}\lambda_t \|x_{t+1} - y_t\|_p^p.$$

The remaining proof of Theorem 10 and Theorem 11 follow slightly different pathways and can be found in Appendix B.

3. Line-Search Free Method for Non-Euclidean Acceleration

In this section, we describe our main algorithm which eliminates the line-search required when implementing Algorithm 1 and works with a more general oracle. We first define our oracle.

Definition 13 *An oracle $\mathcal{O}_{\sigma,p} : \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}_+$ is a* σ-Generalized Monteiro-Svaiter *with respect to p (σ-GMS$_p$) oracle for a function $f : \mathbb{R}^d \to \mathbb{R}$ and $\sigma \in [0, 1)$ if, for every $y \in \mathbb{R}^d$, $(x, \gamma) = \mathcal{O}_{\sigma,p}(y)$ satisfies*

$$\langle \nabla f(x), x - y \rangle \leq -(1 - \sigma)\gamma \|x - y\|_p^p, \quad \text{and} \quad \|\nabla f(x)\|_{p^*} \leq (1 + \sigma)\gamma \|x - y\|_p^{p-1}.$$

Additionally, we say $\mathcal{O}_{\sigma,p}$ satisfies an (s, μ) movement bound *if*

$$\|x - y\|_p \geq \begin{cases} (\gamma/\mu^s)^{1/(s-p)} & \text{if } s < \infty \\ 1/\mu & \text{if } s = \infty. \end{cases}$$

In Appendix C, we show that σ-*GMS*$_p$ oracles generalize ℓ_p^s oracles as well as the Monteiro-Svaiter (MS) oracles from Monteiro and Svaiter (2013); Carmon et al. (2022). In this section we provide the following results.

Theorem 14 *Algorithm 2, given $s > p \geq 2$, $\epsilon > 0$, $x_0 \in \mathbb{R}^d$, $R = O(\|x_0 - x^*\|_p)$, and a σ-GMS$_p$ oracle satisfying an (s, μ) movement bound, returns $\tilde{x} \in \mathbb{R}^d$ such that $f(\tilde{x}) - f(x^\star) \leq \epsilon$ after*

$$O\Big((\mu^s \|x_0 - x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}}\Big)$$

iterations. Each iteration can be implemented in $O(d)$ time plus time of the σ-GMS$_p$ oracle.

Theorem 15 *Algorithm 2, given $p \geq 2$, $\epsilon > 0$, $x_0 \in \mathbb{R}^d$, $R = O(\|x_0 - x^*\|_p)$, and a σ-GMS$_p$ oracle satisfying an (∞, μ) movement bound, returns x_T such that $f(x_T) - f(x^\star) \leq \epsilon$ after*

$$O\left((\mu p \|x_0 - x^\star\|_p)^{\frac{p}{p+1}} \log \frac{\|x_0 - x^\star\|_p}{\epsilon}\right)$$

iterations. Each iteration can be implemented in $O(d)$ time plus time of the σ-GMS$_p$ oracle.

Algorithm 2 General Proximal Point Algorithm (Line-Search Free)

Initialize: $s > p \geq 2$, $a_0 = 1$, $A_0 = 0$, $\bar{\lambda}_1 = 1$, $z_0 = y_0 = x_0 \in \mathbb{R}^d$, $T \geq 1$, $T_{\text{ref}} = 1$, $R > 0$, $\mu > 0$, $\sigma \in [0, 1)$

for $t = 0, \cdots, T-1$ **do**

  if $t \geq 1$ **then**

    if $A_t \geq 2A_{T_{\text{ref}}}$ **then** $T_{\text{ref}} = t$;

    $$\bar{\lambda}_{t+1} = \begin{cases} A_{T_{\text{ref}}}^{-\frac{(s-p)(p+1)}{ps-p+s}} R^{\frac{p(s-p)}{ps-p+s}} \mu^{\frac{sp^2}{s-p+ps}} & \text{if } s < \infty \\ \frac{\mu^{\frac{p^2}{p+1}} R^{\frac{p}{p+1}}}{A_{T_{\text{ref}}}} & \text{if } s = \infty \end{cases}$$

  end

  $\bar{\lambda}_{t+1}(\frac{3+3\sigma}{1-\sigma} a_{t+1})^p = A_{t+1}'^{p-1}$, $A'_{t+1} = A_t + a_{t+1}$ $y_t \leftarrow \frac{A_t}{A_{t+1}} x_t + \frac{a_{t+1}}{A_{t+1}} z_t$

  $(x'_{t+1}, \lambda_{t+1}) \leftarrow \sigma$-*GMS*$_p(y_t)$

  $\beta_{t+1} \leftarrow \min\{1, \frac{\bar{\lambda}_{t+1}}{\lambda_{t+1}}\}$

  $z_{t+1} \leftarrow \arg\min_{z \in \mathbb{R}^d} \sum_{i=1}^{t+1} a_i \beta_i \nabla f(x'_i)^\top (z - y_i) + \omega_p(z, y_0)$

  $x_{t+1} \leftarrow \frac{(1-\beta_{t+1}) A_t x_t + \beta_{t+1} A'_{t+1} x'_{t+1}}{A_t + \beta_{t+1} a_{t+1}}$

  $A_{t+1} \leftarrow A_t + \beta_{t+1} a_{t+1}$

end

return x_T

The proofs of the above results follow from the following potential analysis, which differs from those in Section 2 in that it now accounts for the progress differently depending on whether $\bar{\lambda}_{t+1} < \lambda_{t+1}$ or $\bar{\lambda}_{t+1} \geq \lambda_{t+1}$.

Lemma 16 *The iterates of Algorithm 2 satisfy*

$$A_{t+1}(f(x_{t+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \leq A_t(f(x_t) - f(x^\star)) + \omega_p(x^\star, z_t) - \mathbb{1}_{\{\bar{\lambda}_{t+1} < \lambda_{t+1}\}} \frac{A'_{t+1}(1-\sigma)\bar{\lambda}_{t+1}}{3} \|x'_{t+1} - y_t\|_p^p,$$

and, $A_{t+1}^{1/p} \geq A_t^{1/p} + \mathbb{1}_{\{\bar{\lambda}_{t+1} \geq \lambda_{t+1}\}} \frac{1}{\frac{6+6\sigma}{1-\sigma}\bar{\lambda}_{t+1}^{1/p}}$.

We defer the proofs of both the potential analysis and our main results in this section to Appendix C.

4. Applications

In this section, we cover different applications of our acceleration framework from Section 3. In particular, we apply our framework to ℓ_s-norm regression, where $s \geq p \geq 2$, higher-order smooth optimization, and implement an ℓ_p-ball oracle.

ℓ_s-Norm Regression. We give an algorithm for ℓ_s-regression, where $s \geq p \geq 2$ using an oracle that solves problems of the form,

$$\min_{x \in \mathbb{R}^d} g^\top x + \|Rx\|_2^2 + \|Wx\|_p^p, \tag{2}$$

for any vector $g \in \mathbb{R}^d$ and matrices $R \in \mathbb{R}^{n \times d}$ and $W \in \mathbb{R}^{n \times d}$, where $n \geq d$.

We will prove Theorem 2 which we restate here.

Theorem 2 (From ℓ_s-Regression to Smoothed ℓ_p-Regression) *There is an algorithm that, given $\epsilon > 0$, $s > p \geq 2$, computes an ϵ-approximate solution to ℓ_s-regression (Definition 1) in at most $O(s \cdot n^{\frac{\nu}{1+\nu}} \log^2 \frac{n}{\epsilon})$ iterations for $\nu := \frac{1}{p} - \frac{1}{s}$, each of which can be implemented in $O(d)$ time plus the time needed to solve $\tilde{O}(1)$[3] smoothed ℓ_p-regression problems of the form*

$$\min_{x \in \mathbb{R}^d} g^\top x + \|\tilde{A}x - \tilde{b}\|_2^2 + \|Cx - d\|_p^p .$$

We present the proof in Appendix D. The proof follows from an application of Algorithm 2 for the $\ell_p^s(\lambda)$-proximal oracle (Theorem 4), which reduces solving the ℓ_s-regression problem to $\approx n^{\frac{\nu}{1+\nu}}$ calls to an ℓ_p^s-proximal oracle. We further prove that every such call to the ℓ_p^s-proximal oracle can be solved using $\widetilde{O}(1)$ smoothed ℓ_p-norm problems, i.e., Eq. (2) (Lemma 42).

High-Order Smooth Optimization in General Norms. As an extension of our results, we consider a natural relaxation of the generalized proximal oracle based on regularized high-order Taylor approximations (Bullins, 2020; Gasnikov et al., 2019; Monteiro and Svaiter, 2013; Nesterov and Polyak, 2006; Nesterov, 2008). Namely, letting

$$\mathcal{T}_f^q(y; x) := f(x) + \sum_{i=1}^{q} \frac{1}{i!} \nabla^i f(x)[y - x]^{i-1}$$

denote, for $q \geq 1$, the q^{th}-order Taylor expansion of f at x, we define the following notion of high-order smoothness, which naturally generalizes those in previous works (Nesterov, 2008; Gasnikov et al., 2019).

3. We use $\tilde{O}$ to hide p, s, and $\mathrm{poly}\log n$ factors.

Definition 17 *We say f is q^{th}-order L_q smooth with respect to $\|\cdot\|_p$ if, for all $x, y \in \mathbb{R}^d$,*

$$\left\|\nabla f(y) - \nabla_y \mathcal{T}_f^q(y; x)\right\|_{p^*} \leq \frac{L_q}{q!}\|y - x\|_p^q. \tag{3}$$

We now consider the problem of minimizing a convex function f given what we call a (p, s)-*Taylor oracle*, which, for a centering point c, returns the solution of the regularized problem

$$\min_{x \in \mathbb{R}^d} \mathcal{T}_f^{s-1}(x; c) + \frac{L_{s-1}}{(s-1)!}\|x - c\|_p^s.$$

In contrast to the $\ell_p^s(\lambda)$-proximal oracle, here we have replaced $f(x)$ with its $(s-1)^{th}$-order counterpart $\mathcal{T}_f^{s-1}(x; c)$, and the smoothness parameter now fulfills the role of λ (which was previously a parameter of the oracle). As a natural consequence of our results from Section 3, combined with the observation that that our Taylor oracle implements an $O(1)$-*GMS*$_p$ oracle, we provide the following convergence guarantees for Taylor oracle-based algorithms.

Theorem 9 (High-order Optimization) *For $s > p \geq 2$, $q = s - 1$, let f be q^{th}-order L_q smooth with respect to $\|\cdot\|_p$, $x_0 \in \mathbb{R}^d$, and $\epsilon > 0$. Algorithm 2, implementing an $O(1)$-GMS$_p$ oracle by a (p, s)-Taylor oracle, finds $\tilde{x}$ such that $f(\tilde{x}) - f(x^\star) \leq \epsilon$ in*

$$O_{p,s}\left(\|x_0 - x^*\|_p^{\frac{1}{p(1+\nu)}} (L_q/\epsilon)^{\frac{1}{s(1+\nu)}}\right)$$

iterations. The cost of each iteration is at most $O(d)$ plus the cost of the (p, s)-Taylor oracle.

We would note Theorem 9 obtains the known optimal rates in the case of $p = 2$ and $s > 2$ (Arjevani et al., 2019; Gasnikov et al., 2019), and thus our guarantees allow us to generalize these previous results to all $s > p \geq 2$.

Implementing the ℓ_p Ball Oracle Under Hessian Stability. As in the case of the Euclidean ball oracle (Carmon et al., 2020b; Karimireddy et al., 2018), we rely on the following notion of *Hessian stability* for implementing our more general ℓ_p ball oracles.

Definition 18 (Carmon et al. (2020b), Definition 7) *A twice-differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is (r, ξ)*-Hessian stable with respect to $\|\cdot\|$, *for $r, \xi > 0$, if for all $x, y \in \mathbb{R}^d$ such that $\|x - y\| \leq r$, we have $\xi^{-1}\nabla^2 f(y) \leq \nabla^2 f(x) \leq \xi\nabla^2 f(y)$.*

As its name suggests, this type of stability provides control over *local* changes in the Hessian, and objectives such logistic regression and the softmax loss exhibit favorable Hessian stability with respect to $\|\cdot\|_2$ and $\|\cdot\|_\infty$, respectively (Bach, 2010; Carmon et al., 2020b; Karimireddy et al., 2018). We may further observe that such control entails beneficial (local) conditioning (i.e., strong convexity and smoothness) of f *with respect to* $\|\cdot\|_{\nabla^2 f(y)}$ (where we let $\|u\|_{\nabla^2 f(y)}^2 = u^\top \nabla^2 f(y) u$), and it can be shown that this condition holds when f is α*-quasi-self-concordant with respect to* $\|\cdot\|$, meaning that, for all $x, u, v \in \mathbb{R}^d$, $|\nabla^3 f(x)[v, u, u]| \leq \alpha\|v\|\|u\|_{\nabla^2 f(x)}^2$. It is natural, then, that we may run an appropriate descent method (in terms of $\|\cdot\|_{\nabla^2 f(y)}$, e.g., (Carmon et al., 2020b), Theorem 9), constrained to the region of stability.

We additionally rely on a relaxation of the ℓ_p ball oracle (Definition 6), which allows for a certain amount of approximation in a manner similar to that in Carmon et al. (2020b).

Definition 19 *Given $p \geq 2$, $\delta, r > 0$, an $\ell_p^\infty(\delta, r)$-proximal oracle (also referred to as an* approximate ball-constrained oracle*) for a function $f : \mathbb{R}^d \to \mathbb{R}$ is an oracle that, when queried at a centering point $c \in \mathbb{R}^d$, returns $\tilde{x} \in \mathbb{R}^d$ such that $\left\|\tilde{x} - x^*_{c,r}\right\|_p \leq \delta$, where $x^*_{c,r}$ is solution of the problem*

$$\min_{x \in \mathbb{R}^d : \|x-c\|_p \leq r} f(x).$$

The following theorem establishes that, under an appropriate choice of δ, such an oracle can implement a σ-GMS$_p$ oracle, and its proof can be found in Appendix D.2.

Theorem 20 *Let f be L-smooth w.r.t. $\|\cdot\|_p$, and let $\tilde{x} \in \mathbb{R}^d$ be given by an $\ell_p^\infty(\delta, r)$-proximal oracle for f queried at $c \in \mathbb{R}^d$, for $p \geq 2$, $r, \tilde{\epsilon}, \sigma > 0$, $\min\{\frac{r\sigma}{4}, \frac{\tilde{\epsilon}\sigma}{4rL}\} \geq \delta > 0$. Then, if $\|\nabla f(\tilde{x})\| \geq \tilde{\epsilon}$, an oracle that returns $(\tilde{x}, \tilde{\gamma})$ for $\tilde{\gamma} = \frac{\|\nabla f(\tilde{x})\|}{\|\tilde{x}-y\|_p^{p-1}}$ is a σ-GMS$_p$ oracle, and furthermore it satisfies an $(\infty, (r-\delta)^{-1})$ movement bound.*

Algorithm 3 Constrained Accelerated Newton Descent

Initialize: $a_0 = 1$, $A_0 = 0$, $x_0 = y_0 = z_0 = x_{\text{init}} \in \mathbb{R}^d$, $r > 0$, $T \geq 1$
for $t = 0, \cdots, T-1$ **do**
  $5a_{t+1}^2 = A_{t+1}$, $A_{t+1} = A_t + a_{t+1}$
  $y_t \leftarrow \frac{A_t}{A_{t+1}} x_t + \frac{a_{t+1}}{A_{t+1}} z_t$
  $x_{t+1} \leftarrow \arg\min_{x:\|x-c\| \leq r} \{\langle \nabla f(y_t),\, x - y_t \rangle + \xi \|x - y_t\|^2_{\nabla^2 f(c)}\}$
  $z_{t+1} \leftarrow \arg\min_z \sum_{i \in [t+1]} a_i \nabla f(x_i)^\top (z - y_i) + \|z - y_0\|^2_{\nabla^2 f(c)}$
end
return x_T

Algorithm 4 Constrained Accelerated Newton Descent + Restarting

Initialize: $\delta > 0$, $x_0 \in \mathbb{R}^d$, $r > 0$, $K = O(\log(\frac{\xi dLr}{\mu\delta}))$
for $k = 0, \cdots, K-1$ **do**
  $x_{k+1} =$ Constrained Accelerated Newton Descent$(x_{\text{init}} = x_k, r, T = O(\xi))$
end
return x_K

It remains to show how we may implement an $\ell_p^\infty(\delta, r)$-proximal oracle for Hessian stable functions, which the following theorem provides.

Theorem 21 *Let f be L-smooth and μ-strongly convex w.r.t. $\|\cdot\|_2$, and (r, ξ)-Hessian stable w.r.t. $\|\cdot\|_p$. Then, there is an algorithm (Algorithm 4) that implements an $\ell_p^\infty(\delta, r)$-proximal oracle, using $O(\xi \log(\frac{\xi dLr}{\mu\delta}))$ queries to a gradient oracle and one query to a Hessian oracle of f.*

The proof (found in Appendix D.3) closely follows that of Theorem 10, along with a standard restarting argument (e.g., (Roulet and d'Aspremont, 2017)) to attain linear convergence rates when paired with the strong convexity that comes from Hessian stability.

5. Lower Bounds

In this section, we complement our upper bound results with matching lower bounds, for both $s < \infty$ and $s = \infty$ cases, for algorithms that are *zero-respecting* (Carmon et al., 2020a). This condition, which entails, roughly speaking, that the support of any iterate is restricted by the span of all preceding iterates, plays an important role in establishing lower bounds for a variety of optimization settings (Arjevani et al., 2023; Carmon et al., 2021a).

When $s < \infty$: Proximal Point Oracles. We begin by proving lower bounds for zero-respecting proximal point algorithms when minimizing a function f using an $\ell_p^s(\lambda)$-proximal oracle, where we let $\mathcal{O}_{f,\lambda,p,s}(c)$ denote the result upon querying the oracle at $c \in \mathbb{R}^d$.

We first introduce an appropriate notion of *zero-respecting algorithms* for our setting. Namely, letting $\mathrm{supp}\{\boldsymbol{v}\} := \{i \in [d] : \boldsymbol{v}_i \neq 0\}$ for $\boldsymbol{v} \in \mathbb{R}^d$, we say a sequence $x^{(0)}, x^{(1)}, \ldots$ is $\ell_p^s(\lambda)$-*proximal zero-respecting with respect to* f if

$$\mathrm{supp}\left\{x^{(t)}\right\} \subseteq \bigcup_{t'<t} \mathrm{supp}\left\{\mathcal{O}_{f,\lambda,p,s}(x^{(t')})\right\} \text{ for each } t \in \mathbb{N}.$$

We may then naturally define an $\ell_p^s(\lambda)$-*proximal zero-respecting* algorithm.

Definition 22 ($\ell_p^s(\lambda)$-Proximal Zero-Respecting Algorithm) *We say an algorithm $\mathcal{A}$ is* $\ell_p^s(\lambda)$-proximal zero-respecting *if, for any $f : \mathbb{R}^d \to \mathbb{R}$, the iterate sequence $\mathcal{A}[f]$ is zero-respecting with respect to f.*

We further consider the following proximal-type notion of a *zero-chain*.

Definition 23 ($\ell_p^s(\lambda)$-Proximal Zero-Chain) *A function $f : \mathbb{R}^d \to \mathbb{R}$ is an* $\ell_p^s(\lambda)$-proximal zero-chain *if, for every $x \in \mathbb{R}^d$, $\mathrm{supp}\{x\} \subseteq \{1, \ldots, i-1\}$ implies $\mathrm{supp}\{\mathcal{O}_{f,\lambda,p,s}(x)\} \subseteq \{1, \ldots, i\}$.*

We would note that, while previous lower bounds for smooth convex optimization (Arjevani et al., 2019; Nemirovskij and Yudin, 1983) hold for algorithms that can generate approximate proximal point oracle updates (up to polylogarithmic error), here we provide lower bounds for when we have an *exact* proximal point oracle. We now state our main lower bound theorem for this section, where we recall that $x^* := \arg\min_{x \in \mathbb{R}^d} g_k(x)$.

Theorem 5 ($\ell_p^s(\lambda)$-Proximal Oracle Optimization Lower Bound) *For every $\ell_p^s(\lambda)$-proximal zero-respecting algorithm (Definition 22) given $p, s \geq 2$, $\lambda > 0$, $\epsilon > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\lambda \|x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}})$ iterations of the algorithm satisfies $f(x) - f(x^\star) \geq \epsilon$.*

While we provide the full proof of Theorem 5 in Appendix E.3, the key construction is based on a scaled instance of Nemirovski's function (Nemirovskij and Yudin, 1983), whose parameters depend on properties (i.e., p, s, λ) defining the proximal oracle.

Definition 24 (Scaled Nemirovski instance) *Let $k, d \in \mathbb{N}$ be such that $1 \leq k \leq d$, and let $R > 0$. We define*

$$f_k(x) = \beta_k \max_{1 \leq i \leq k} \{x_i - i \cdot \alpha_k\},$$

where

$$\beta_k := \left(\frac{s-1}{s}\right)^{2(s-1)} \frac{s\lambda R^{s-1}}{(k+1)^{\frac{(p+1)(s-1)}{p}}},\ \alpha_k := \frac{s}{s-1}\left(\frac{\beta_k}{s\lambda}\right)^{\frac{1}{s-1}} = \frac{(s-1)R}{s(k+1)^{\frac{p+1}{p}}}.$$

Following the approach as described in previous works (Carmon et al., 2020b; Diakonikolas and Guzmán, 2019), we may handle unconstrained domains by considering the following problem:

$$\min_{x\in R^d} g_k(x) := \max\Big\{f_k(x), \beta_k(\|x\|_p - 2R) - \alpha_k\Big\}. \tag{4}$$

The following lemma, which is proved in Appendix E, shows that the iterates produced by our $\ell_p^s(\lambda)$-proximal oracle satisfy the required condition on their support.

Lemma 25 *Let $k, d \in \mathbb{N}$ be such that $1 \le k \le d$, and let g_k be as in (4). Then, g_k is an $\ell_p^s(\lambda)$-proximal zero-chain.*

When $s = \infty$: Ball Oracle. For this case, we consider the ℓ_p ball oracle, i.e.,

$$\mathcal{O}_{f,r,p,\infty}(c) = \arg\min_{\|x-c\|_p \le r} f(x).$$

Similar to the previous section, we may define analogous notions of $\ell_p^\infty(r)$-*proximal zero-respecting algorithms* and an $\ell_p^\infty(r)$-*proximal zero-chain*. We say a sequence $x^{(0)}, x^{(1)}, \dots$ is $\ell_p^\infty(r)$-*proximal zero-respecting with respect to* f if

$$\operatorname{supp}\Big\{x^{(t)}\Big\} \subseteq \bigcup_{t'<t} \operatorname{supp}\Big\{\mathcal{O}_{f,r,p,\infty}(x^{(t')})\Big\} \text{ for each } t \in \mathbb{N},$$

We may then similarly define a $\ell_p^\infty(r)$-*proximal zero-respecting* algorithm.

Definition 26 ($\ell_p^\infty(r)$-Proximal Zero-Respecting Algorithm) *We say an algorithm $\mathcal{A}$ is $\ell_p^\infty(r)$-*proximal zero-respecting *if, for any $f : \mathbb{R}^d \to \mathbb{R}$, the iterate sequence $\mathcal{A}[f]$ is $\ell_p^\infty(r)$-proximal zero-respecting with respect to f.*

We further consider the following ball oracle-type notion of a $\ell_p^\infty(r)$-*zero-chain*.

Definition 27 ($\ell_p^\infty(r)$-Proximal Zero-Chain) *A function $f : \mathbb{R}^d \to \mathbb{R}$ is an $\ell_p^\infty(r)$-*proximal zero-chain *if, for every $x \in \mathbb{R}^d$, $\operatorname{supp}\{x\} \subseteq \{1, \dots, i-1\}$ implies $\operatorname{supp}\{\mathcal{O}_{f,r,p,\infty}(x)\} \subseteq \{1, \dots, i\}$.*

We now state the main result (Theorem 8) of this section, whose proof can be found in Appendix E.5.

Theorem 8 (ℓ_p Ball-constrained Oracle Optimization Lower Bound) *For every $\ell_p^\infty(r)$-proximal zero-respecting algorithm (Definition 26) given $p \ge 2$, $r > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\|x^\star\|_p/r)^{\frac{p}{p+1}})$ iterations of the algorithm satisfies*

$$f(x) - f(x^\star) \ge \Omega(\|x^\star\|_p^{1/(p+1)} r^{p/(p+1)}).$$

Acknowledgments

Thank you to anonymous reviewers for your feedback. DA is supported by Dr. Max Rössler, the Walter Haefner Foundation and the ETH Zürich Foundation. AS was supported in part by a Microsoft Research Faculty Fellowship, NSF CAREER Grant CCF-1844855, NSF Grant CCF-1955039, and a PayPal research award. Part of this work was conducted while the authors were visiting the Simons Institute for the Theory of Computing.

References

Deeksha Adil and Sushant Sachdeva. Faster p-norm minimizing flows, via smoothed q-norm problems. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 892–910. SIAM, 2020.

Deeksha Adil, Rasmus Kyng, Richard Peng, and Sushant Sachdeva. Iterative refinement for ℓ_p-norm regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1405–1424. SIAM, 2019.

Deeksha Adil, Rasmus Kyng, Richard Peng, and Sushant Sachdeva. Fast algorithms for ℓ_p- regression. *J. ACM*, 2024. ISSN 0004-5411. doi: 10.1145/3686794. URL https://doi.org/10.1145/3686794.

Naman Agarwal, Zeyuan Allen-Zhu, Brian Bullins, Elad Hazan, and Tengyu Ma. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1195–1199, 2017.

Morteza Alamgir and Ulrike Luxburg. Phase transition in the family of p-resistances. *Advances in neural information processing systems*, 24, 2011.

Yossi Arjevani, Ohad Shamir, and Ron Shiff. Oracle complexity of second-order methods for smooth convex optimization. *Mathematical Programming*, 178(1):327–360, 2019.

Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199(1):165–214, 2023.

Francis Bach. Self-concordant analysis for logistic regression. 2010.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Sébastien Bubeck, Michael B Cohen, Yin Tat Lee, and Yuanzhi Li. An homotopy method for lp regression provably beyond self-concordance and in input-sparsity time. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1130–1137, 2018.

Sébastien Bubeck, Qijia Jiang, Yin-Tat Lee, Yuanzhi Li, and Aaron Sidford. Complexity of highly parallel non-smooth convex optimization. *Advances in neural information processing systems*, 32, 2019a.

Sébastien Bubeck, Qijia Jiang, Yin Tat Lee, Yuanzhi Li, and Aaron Sidford. Near-optimal method for highly smooth convex optimization. In *Conference on Learning Theory*, pages 492–507. PMLR, 2019b.

Brian Bullins. Highly smooth minimization of non-smooth problems. In *Conference on Learning Theory*, pages 988–1030. PMLR, 2020.

Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020a.

Yair Carmon, Arun Jambulapati, Qijia Jiang, Yujia Jin, Yin Tat Lee, Aaron Sidford, and Kevin Tian. Acceleration with a ball optimization oracle. *Advances in Neural Information Processing Systems*, 33:19052–19063, 2020b.

Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points ii: first-order methods. *Mathematical Programming*, 185(1):315–355, 2021a.

Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Thinking inside the ball: Near-optimal minimization of the maximal loss. In *Conference on Learning Theory*, pages 866–882. PMLR, 2021b.

Yair Carmon, Danielle Hausler, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Optimal and adaptive monteiro-svaiter acceleration. *Advances in Neural Information Processing Systems*, 35: 20338–20350, 2022.

Yair Carmon, Arun Jambulapati, Yujia Jin, Yin Tat Lee, Daogao Liu, Aaron Sidford, and Kevin Tian. Resqueing parallel and private stochastic convex optimization. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 2031–2058. IEEE, 2023.

Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. A whole new ball game: A primal accelerated method for matrix games and minimizing the maximum of smooth functions. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3685–3723. SIAM, 2024.

Flavio Chierichetti, Sreenivas Gollapudi, Ravi Kumar, Silvio Lattanzi, Rina Panigrahy, and David P Woodruff. Algorithms for ℓ_p low-rank approximation. In *International Conference on Machine Learning*, pages 806–814. PMLR, 2017.

Patrick L Combettes and Jean-Christophe Pesquet. A douglas–rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):564–574, 2007.

Juan Pablo Contreras, Cristóbal Guzmán, and David Martinez-Rubio. Non-euclidean high-order smooth convex optimization extended abstract. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 1330–1330. PMLR, 2025.

Jelena Diakonikolas and Cristóbal Guzmán. Lower bounds for parallel and randomized convex optimization. In *Conference on Learning Theory*, pages 1132–1157. PMLR, 2019.

Jelena Diakonikolas and Lorenzo Orecchia. The approximate duality gap technique: A unified theory of first-order methods. *SIAM Journal on Optimization*, 29(1):660–689, 2019.

Abderrahim Elmoataz, Matthieu Toutain, and Daniel Tenbrinck. On the p-laplacian and ∞-laplacian on graphs with applications in image and data processing. *SIAM Journal on Imaging Sciences*, 8 (4):2412–2451, 2015.

Abderrahim Elmoataz, Xavier Desquesnes, and M Toutain. On the game p-laplacian on weighted graphs with applications in image processing and data clustering. *European Journal of Applied Mathematics*, 28(6):922–948, 2017.

Mauricio Flores, Jeff Calder, and Gilad Lerman. Analysis and algorithms for ℓ_p-based semi-supervised learning on graphs. *Applied and Computational Harmonic Analysis*, 60:77–122, 2022.

Roy Frostig, Rong Ge, Sham Kakade, and Aaron Sidford. Un-regularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. In *International Conference on Machine Learning*, pages 2540–2548. PMLR, 2015.

Alexander Gasnikov, Pavel Dvurechensky, Eduard Gorbunov, Evgeniya Vorontsova, Daniil Selikhanovych, César A Uribe, Bo Jiang, Haoyue Wang, Shuzhong Zhang, Sébastien Bubeck, et al. Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In *Conference on Learning Theory*, pages 1392–1393. PMLR, 2019.

Osman Güler. New proximal point algorithms for convex minimization. *SIAM Journal on Optimization*, 2(4):649–664, 1992.

Cristóbal Guzmán and Arkadi Nemirovski. On lower complexity bounds for large-scale smooth convex optimization. *Journal of Complexity*, 31(1):1–14, 2015.

Arun Jambulapati, Yang P Liu, and Aaron Sidford. Improved iteration complexities for overconstrained p-norm regression. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 529–542, 2022.

Arun Jambulapati, Aaron Sidford, and Kevin Tian. Closing the computational-query depth gap in parallel stochastic convex optimization. In *Conference on Learning Theory*. PMLR, 2024.

Sai Praneeth Karimireddy, Sebastian U Stich, and Martin Jaggi. Global linear convergence of newton's method without strong-convexity or lipschitz gradients. *arXiv preprint arXiv:1806.00413*, 2018.

Narendra Karmarkar. A new polynomial-time algorithm for linear programming. In *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*, pages 302–311, 1984.

Rasmus Kyng, Anup Rao, Sushant Sachdeva, and Daniel A Spielman. Algorithms for lipschitz learning on graphs. In *Conference on Learning Theory*, pages 1190–1223. PMLR, 2015.

Yin Tat Lee, Zhao Song, and Qiuyi Zhang. Solving empirical risk minimization in the current matrix multiplication time. In *Conference on Learning Theory*, pages 2140–2157. PMLR, 2019.

Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. A universal catalyst for first-order optimization. *Advances in Neural Information Processing Systems*, 28, 2015.

Haihao Lu, Robert M Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

Renato DC Monteiro and Benar Fux Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2):1092–1125, 2013.

Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

Yu Nesterov. Accelerating the cubic regularization of newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

Yurii Nesterov. A method for solving the convex programming problem with convergence rate o (1/k2). In *Dokl akad nauk Sssr*, volume 269, page 543, 1983.

Yurii Nesterov. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, 186:157–183, 2021.

Yurii Nesterov and Arkadii Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical programming*, 108(1):177–205, 2006.

Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and trends® in Optimization*, 1 (3):127–239, 2014.

James Renegar. A polynomial-time algorithm, based on newton's method, for linear programming. *Mathematical programming*, 40(1):59–93, 1988.

R Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

Vincent Roulet and Alexandre d'Aspremont. Sharpness, restart and acceleration. *Advances in Neural Information Processing Systems*, 30, 2017.

Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162:83–112, 2017.

Shai Shalev-Shwartz and Tong Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *International Conference on Machine Learning*, pages 64–72. PMLR, 2014.

Stephen J Wright. *Primal-dual interior-point methods*. SIAM, 1997.

C Zălinescu. On uniformly convex functions. *Journal of Mathematical Analysis and Applications*, 95(2):344–374, 1983.

Appendix A. Preliminaries

Definition 28 (Bregman Divergence) *For a strictly convex, continuously differentiable $d : D \to \mathbb{R}_{\geq 0}$, we define its Bregman divergence between $x \in D$ and $y \in D$ as*

$$\omega(x, y) := d(x) - d(y) - \langle \nabla d(y), x - y \rangle.$$

Note that ω_p is the Bregman divergence of the function $d(x) = \|x - x_0\|_p^p$ for an initial point x_0.

Lemma 29 (Three-Point Property) *For any $x, y, z \in D$,*

$$\omega(x, y) = \omega(x, z) + \omega(z, y) - (x - z)^\top (\nabla d(y) - \nabla d(z)).$$

We also use the following result, which can be found in, e.g., Zălinescu (1983).

Lemma 30 (Lemma 3.1 Zălinescu (1983)) *For any x, y and $p \geq 2$, $\omega_p(y, x) \geq \frac{1}{2^{p+1}} \|y - x\|_p^p$.*

We use the following fact about the gradient and Hessian of the ℓ_p-norm function repeatedly.

Fact 31 *The gradient and Hessian of the function $\|Ax\|_p^p$, for any $p \geq 2$, and matrix $A \in \mathbb{R}^{n \times d}$ and vector $x \in \mathbb{R}^d$ is,*

$$\nabla_x \|Ax\|_p^p = p \cdot \textsc{Diag}\big(|Ax|^{p-2}\big) Ax, \quad \nabla_{xx}^2 \|Ax\|_p^p = p(p-1) A^\top \textsc{Diag}\big(|Ax|^{p-2}\big) A.$$

Appendix B. Proofs for Conceptual Method

B.1. Proof of Lemma 12

Proof We start by considering the left-hand side of the inequality

$$\begin{aligned}
&A_{t+1}(f(x_{t+1}) - f(x^\star)) - A_t(f(x_t) - f(x^\star)) \\
&\quad = a_{t+1}(f(x_{t+1}) - f(x^\star)) + A_t(f(x_{t+1}) - f(x_t)) \\
&\quad \leq a_{t+1}\langle \nabla f(x_{t+1}), x_{t+1} - x^\star \rangle + A_t \langle \nabla f(x_{t+1}), x_{t+1} - x_t \rangle \quad \text{(convexity of } f\text{)} \\
&\quad = \langle \nabla f(x_{t+1}), A_{t+1} x_{t+1} - A_t x_t - a_{t+1} x^\star \rangle \\
&\quad = \langle \nabla f(x_{t+1}), A_{t+1} x_{t+1} + a_{t+1} z_t - A_{t+1} y_t - a_{t+1} x^\star \rangle \quad \text{(Since } A_{t+1} y_t = A_t x_t + a_{t+1} z_t\text{)} \\
&\quad = A_{t+1} \underbrace{\langle \nabla f(x_{t+1}), x_{t+1} - y_t \rangle}_{\text{Term 1}} + a_{t+1} \underbrace{\langle \nabla f(x_{t+1}), z_t - x^\star \rangle}_{\text{Term 2}}.
\end{aligned}$$

We first give a bound on Term 2. Using the KKT conditions for z_t, i.e., differentiating the equation defining z_t in the algorithm w.r.t z, we have for all $t \geq 0$,

$$\sum_{i \in [t]} a_i \nabla f(x_i) = -\nabla_z \|z - y_0\|_p^p \Big|_{z = z_t} = -\nabla \omega_p(z_t, y_0) \quad \text{for all} \quad t,$$

where we used that $\omega_p(z_t, y_0) = \|z_t - y_0\|_p^p$ since $x_0 = y_0$. Therefore,

$$a_{t+1} \nabla f(x_{t+1}) = \nabla \omega_p(z_t, y_0) - \nabla \omega_p(z_{t+1}, y_0).$$

Replacing $a_{t+1}\nabla f(x_{t+1})$ with the above in Term 2, we get

$$\begin{aligned}&a_{t+1}\langle \nabla f(x_{t+1}), z_t - x^\star\rangle = a_{t+1}\langle \nabla f(x_{t+1}), z_{t+1} - x^\star\rangle + a_{t+1}\langle \nabla f(x_{t+1}), z_t - z_{t+1}\rangle \\ =&\langle \nabla \omega_p(z_t, y_0) - \nabla \omega_p(z_{t+1}, y_0), z_{t+1} - x^\star\rangle + a_{t+1}\langle \nabla f(x_{t+1}), z_t - z_{t+1}\rangle.\end{aligned}$$

Applying the Bregman three-point property (Lemma 29) that yields that

$$a_{t+1}\langle \nabla f(x_{t+1}), z_t - x^\star\rangle = \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) - \omega_p(z_t, z_{t+1}) + a_{t+1}\langle \nabla f(x_{t+1}), z_t - z_{t+1}\rangle\,.$$

We now apply Young's inequality to get

$$\begin{aligned}a_{t+1}\langle \nabla f(x_{t+1}), z_t - x^\star\rangle \le \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) - \omega_p(z_t, z_{t+1}) \\ + \frac{p-1}{p}\|2a_{t+1}\nabla f(x_{t+1})\|_{p^*}^{p^*} + \frac{1}{p2^p}\|z_t - z_{t+1}\|_p^p.\end{aligned}$$

Since $\omega_p(x, y) \ge \frac{1}{2^{p+1}}\|x - y\|_p^p$ (Lemma 30), we obtain the following bound on Term 2,

$$a_{t+1}\langle \nabla f(x_{t+1}), z_t - x^\star\rangle \le \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) + 4a_{t+1}^{p^*}\frac{p-1}{p}\|\nabla f(x_{t+1})\|_{p^*}^{p^*}. \tag{5}$$

We next bound Term 1. First, we observe by the optimality guarantees of x_{t+1} that

$$\nabla f(x_{t+1}) = -\lambda_t p \cdot \textsc{Diag}(|x_{t+1} - y_t|)^{p-2}(x_{t+1} - y_t),$$

where the above derivative is computed using Fact 31. Therefore,

$$\langle \nabla f(x_{t+1}), x_{t+1} - y_t\rangle = -\lambda_t\|x_{t+1} - y_t\|_p^p. \tag{6}$$

Combining the bounds on Term 1 (5) and Term 2 (6) yield

$$\begin{aligned}&A_{t+1}(f(x_{t+1}) - f(x^\star)) - A_t(f(x_t) - f(x^\star)) \\ \le& - A_{t+1}\langle \nabla f(x_{t+1}), x_{t+1} - y_t\rangle + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) + 4a_{t+1}^{p^*}\frac{p-1}{p}\|\nabla f(x_{t+1})\|_{p^*}^{p^*} \\ \le& - A_{t+1}\lambda_t\|x_{t+1} - y_t\|_p^p + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) + 4a_{t+1}^{p^*}\lambda_t^{p^*}\cdot\frac{p-1}{p}\|x_{t+1} - y_t\|_p^p\end{aligned}$$

Now, since $A_{t+1}^{p-1} = 5^{p-1}\lambda_t a_{t+1}^p$, the above becomes

$$A_{t+1}(f(x_{t+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \le A_t(f(x_t) - f(x^\star)) + \omega_p(x^\star, z_t) - A_{t+1}\lambda_t\|x_{t+1} - y_t\|_p^p.$$

■

B.2. Proof of Theorem 10

We need to prove the existence of λ_t $(t \ge 0)$ satisfying the conditions of Algorithm 1, using a similar continuity argument as in Lemma 3.2 of Bubeck et al. (2019b).

Lemma 32 *Let $A \ge 0, x, y \in \mathbb{R}^d$ such that $f(x) \ne f(x^\star)$. Also let $z(x)$ be a continuous function of x. Define the following functions:*

- $a(\lambda)$ *is the solution of the equation* $\lambda^{\frac{1}{p-1}} a(\lambda)^{p^*} - a(\lambda) - A = 0$.
- $y(\lambda) = \frac{a(\lambda)}{A+a(\lambda)} z(x) + \frac{A}{A+a(\lambda)} x$
- $x(\lambda) = \arg\min_{\boldsymbol{w}} f(\boldsymbol{w}) + \lambda \|\boldsymbol{w} - y(\lambda)\|_p^p$
- $g(\lambda) = \frac{\|x(\lambda)-y(\lambda)\|_p^{s-p}}{\lambda}$

Then, for every $z \in \mathbb{R}_+$ *there exists* $w \in \mathbb{R}_+$ *such that* $g(w) = z$.

Proof We first claim that $g(\lambda)$ is a continuous function of λ. It is easy to see that $y(\lambda)$ is continuous. The continuity of $x(\lambda)$ follows from the fact that since f is convex, $f(\boldsymbol{w}) + \lambda\|\boldsymbol{w} - c\|_p^p$ is strictly convex and there is a unique minimizer for every λ. Now, $x \neq y$ since $f(x) \neq f(x^\star)$. Therefore, $g(0) = \infty$ and $g(\infty) = 0$, which concludes the proof. ■

The following lemma provides a means of lower bounding A_t for all $t \geq 1$.

Lemma 33 *For all* $t \geq 1$, *Algorithm 1 guarantees that*

$$A_{t+1}^{1/p} \geq A_t^{1/p} + \frac{1}{5p\lambda_t^{1/p}} \text{ and, consequently, } A_t^{1/p} \geq A_0^{1/p} + \sum_{t\in[T]} \frac{1}{5p\lambda_t^{1/p}}.$$

Proof We know from Algorithm 1 that $A_{t+1} = A_t + a_{t+1}$ and $a_{t+1}^p = \frac{A_{t+1}^{p-1}}{5^{p-1}\lambda_t}$. Therefore,

$$A_t = A_{t+1} - \left(\frac{A_{t+1}^{p-1}}{5^{p-1}\lambda_t}\right)^{1/p} = A_{t+1}\left(1 - \frac{1}{(5^{p-1}\lambda_t A_{t+1})^{1/p}}\right).$$

We will use that for any $x > -1$ and $r > 1$, $(1+x)^r \geq 1 + rx$. Observe that $\frac{1}{(5^{p-1}\lambda_t A_{t+1})^{1/p}} = \frac{a_{t+1}}{A_{t+1}} < 1$. Applying this inequality for $r = p$ and $x = -\frac{1}{(5^{p-1}\lambda_t A_{t+1})^{1/p}}$, we get

$$A_t \leq A_{t+1}\left(1 - \frac{1}{(5^{p-1}p\lambda_t A_{t+1})^{1/p}}\right)^p,$$

which on rearranging gives us

$$A_t^{1/p} + \frac{1}{5p\lambda_t^{1/p}} \leq A_{t+1}^{1/p}.$$

■

Before we prove our main result, we state a result from Carmon et al. (2022) which is useful in our final proof.

Lemma 34 (Lemma 3, Carmon et al. (2022)) *Let* $B_1, \cdots, B_k \in \mathbb{R}_{>0}$, $r_1, \cdots, r_k \in \mathbb{R}_{>0}$ *satisfy* $B_i^m \geq \beta \sum_{j\in[i]} B_j r_j$ *for some* $m > 1$, $\beta > 0$, *and all* $i \in [k]$. *Then* $B_i \geq (\frac{m-1}{m}\beta \cdot \sum_{j\in[i]} r_j)^{\frac{1}{m-1}}$ *for all* $i \in [k]$.

We are now ready to prove the main result of Section 2.

Theorem 10 *Let $s < \infty$. Given f, $x_0 \in \mathbb{R}^d$, and $\epsilon > 0$, there exist, for all $t \geq 0$, λ_t, x_{t+1} satisfying the conditions of Algorithm 1, and it outputs x_T such that $f(x_T) - f(x^\star) \leq \epsilon$, in*

$$O\left(\frac{s^{s(1+\nu)}}{p^{s\nu}} \cdot \frac{\lambda \|x_0 - x^*\|_p^s}{\epsilon}\right)^{\frac{1}{s(1+\nu)}} \quad \textit{iterations.}$$

Proof [Proof of Theorem 10] Let $r_t := \|x_{t+1} - y_t\|_p$. Note that $\lambda_t = \lambda r_t^{s-p}$ and therefore $\lambda_t \|x_{t+1} - y_t\|_p^p = \lambda_t^{\frac{s}{s-p}} \lambda^{-\frac{p}{s-p}}$. Since we are using an $\ell_p^s(\lambda)$-proximal oracle, the conditions of Theorem 12 are true. The lemma then implies that for all $T \geq 1$

$$A_T(f(x_T) - f(x^\star)) + \omega_p(x^\star, z_T) \leq A_0(f(x_0) - f(x^\star)) + \omega_p(x^\star, z_0) - \sum_{t\in[T]} A_t \lambda_t^{\frac{s}{s-p}} \lambda^{-\frac{p}{s-p}}. \quad (7)$$

Since $f(x_T) - f(x^\star) \geq 0$ and $\omega_p(x^\star, z_t) \geq 0$ for all t,

$$\lambda^{-\frac{p}{s-p}} \sum_{t\in[T]} A_t \lambda_t^{\frac{s}{s-p}} \leq \omega_p(x^\star, z_0) = \Psi_0$$

From Theorem 33, $A_t^{1/p} \geq \sum_{t\in[T]} \frac{1}{5p\lambda_t^{1/p}}$. Applying Holder's inequality with $\alpha = \frac{1}{\nu}$ yields,

$$\begin{aligned}\sum_{t\in[T]} A_t^{\frac{1}{1+\alpha}} &= \sum_{t\in[T]} A_t^{\frac{1}{1+\alpha}} \lambda_t^{\frac{1}{p(1+\alpha^{-1})}} \lambda_t^{-\frac{1}{p(1+\alpha^{-1})}} \leq \left(\sum_{t\in[T]} A_t \lambda_t^{\frac{s}{s-p}}\right)^{(1+\alpha)^{-1}} \left(\sum_{t\in[T]} \lambda_t^{-\frac{1}{p}}\right)^{(1+\alpha^{-1})^{-1}} \\ &\leq \left(\lambda^{\frac{p}{s-p}} \Psi_0\right)^{\frac{1}{1+\alpha}} \left(5pA_T^{1/p}\right)^{\frac{\alpha}{1+\alpha}}\end{aligned}$$

where we used that $\frac{\alpha}{1+\alpha} = \frac{1}{1+\alpha^{-1}}$ and $(1+\alpha)^{-1} + (1+\alpha^{-1})^{-1} = 1$.

We now use the above to give a lower bound on A_T. Using Lemma 34, for $m = \alpha/p$, $B_j = A_t^{\frac{1}{1+\alpha}}$, $r_j = 1$, $\beta = \left((5p)^\alpha \lambda^{\frac{p}{s-p}} \Psi_0\right)^{-1/(1+\alpha)}$,

$$A_T^{\frac{1}{1+\alpha}} \geq \left(\frac{\alpha - p}{\alpha} \left((5p)^\alpha \lambda^{\frac{p}{s-p}} \Psi_0\right)^{-1/(1+\alpha)} \cdot T\right)^{\frac{p}{\alpha-p}} = \left(\frac{p}{s}\left(\frac{1}{(5p)^\alpha \lambda^{\frac{p}{s-p}} \Psi_0}\right)^{\frac{1}{1+\alpha}} \cdot T\right)^{\frac{s-p}{p}}.$$

Therefore

$$A_T \geq \left(\frac{1}{s}\right)^{\frac{s-p+sp}{p}} p^{\frac{s-p}{p}} \left(\frac{1}{\lambda^{\frac{p}{s-p}} \Psi_0}\right)^{\frac{s-p}{p}} \cdot T^{\frac{ps+s-p}{p}}.$$

Further, using Eq. (7) and $\nu = \frac{1}{p} - \frac{1}{s}$,

$$f(x_T) - f(x^\star) \leq \frac{\Psi_0}{A_T} \leq \left(\frac{s^{\frac{s-p+sp}{p}}}{p^{\frac{s-p}{p}}}\right) \frac{\lambda \Psi_0^{s/p}}{T^{\frac{sp+s-p}{p}}} = \left(\frac{s^{s(1+\nu)}}{p^{s\nu}}\right) \frac{\lambda \Psi_0^{s/p}}{T^{s(1+\nu)}}.$$

The result now follows from noting that $\Psi_0 = \|x^\star - x_0\|_p^p$ ■

B.3. Proof of Theorem 11

Lemma 35 *Let $f : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable and strictly convex. For all $c \in \mathbb{R}^d$, define*

$$x = \text{argmin}_{x \in \mathbb{R}^d | \|x-c\|_p \le r} f(x).$$

Then x is either the global minimizer of f, or $\|x - c\|_p = r$ and

$$\nabla f(x) = -\frac{\|\nabla f(x)\|_{p/(p-1)}}{r^{p-1}} \text{DIAG}(|x-c|)^{p-2}(x-c).$$

Proof The Lagrange dual gives,

$$\min_y \max_\lambda f(y) + \frac{\lambda}{p}\left(r^p - \|y-c\|_p^p\right).$$

There is some $\lambda \ge 0$ such that,

$$\nabla f(x) = -\lambda \text{DIAG}(|x-c|)^{p-2}(x-c).$$

If $\lambda = 0$, then $\nabla f(x) = 0$ and x is the optimizer of f. If $\lambda > 0$, then $\|x-c\|_p = r$, and as a result,

$$\|\nabla f(x)\|_{p/(p-1)} = \lambda \|x-c\|_p^{p-1} = \lambda r^{p-1}.$$

From the above we get, $\lambda = \frac{\|\nabla f(x)\|_{p/(p-1)}}{r^{p-1}}$, concluding the proof. ■

We now state one final result from Carmon et al. (2022) and then prove the main result of the section.

Lemma 36 (Lemma 4, Carmon et al. (2022)) *Let $B_1, \cdots, B_k \in \mathbb{R}_{>0}$ be non-decreasing and, $r_1, \cdots, r_k \in \mathbb{R}_{>0}$ satisfy $B_i \ge \beta \sum_{j \in [i]} B_j r_j$ for some $\beta > 0$, and all $i \in [k]$. Then $B_i \ge \exp\left(\beta \cdot \sum_{j \in [i]} r_j - 1\right) B_1$ for all $i \in [k]$.*

Proof [Proof of Theorem 11]

From Lemma 35, for $\lambda_t = \frac{\|\nabla f(x_{t+1})\|_{p/(p-1)}}{r^{p-1}}$, Lemma 12 holds. Similar to the proof of Lemma 32 we can show that λ_t's exist. Further, using Lemma 12 and the fact that $\|x_{t+1} - y_t\|_p = r$ (Lemma 35), implies that for all $T \ge 1$

$$A_T(f(x_T) - f(x^\star)) + \omega_p(x^\star, z_T) \le A_0(f(x_0) - f(x^\star)) + \omega_p(x^\star, z_0) - r^p \sum_{t \in [T]} A_t \lambda_t. \tag{8}$$

Since $f(x_T) - f(x^\star) \ge 0$ and $\omega_p(x^\star, z_t) \ge 0$ for all t. This implies that

$$r^p \sum_{t \in [T]} A_t \lambda_t \le \omega_p(x^\star, z_0) = \Psi_0$$

Since, $A_t^{1/p} \ge \sum_{t \in [T]} \frac{1}{5p\lambda_t^{1/p}}$ by Lemma 33, applying Holder's inequality for some α yields that

$$\sum_{t \in [T]} A_t^{\frac{1}{1+\alpha}} = \sum_{t \in [T]} A_t^{\frac{1}{1+\alpha}} \lambda_t^{\frac{1}{p(1+\alpha^{-1})}} \lambda_t^{-\frac{1}{p(1+\alpha^{-1})}} \le \left(\sum_{t \in [T]} A_t \lambda_t^{\alpha/p}\right)^{(1+\alpha)^{-1}} \left(\sum_{t \in [T]} \lambda_t^{-\frac{1}{p}}\right)^{(1+\alpha^{-1})^{-1}}$$
$$\le \left(r^{-p}\Psi_0\right)^{\frac{1}{1+\alpha}} \left(5pA_T^{1/p}\right)^{\frac{\alpha}{1+\alpha}}$$

where we used that $\frac{\alpha}{1+\alpha} = \frac{1}{1+\alpha^{-1}}$ and $(1+\alpha)^{-1} + (1+\alpha^{-1})^{-1} = 1$.

We now use the above to give a lower bound on A_T. Using Lemma 36, for $\alpha = p$, $B_j = A_t^{\frac{1}{1+\alpha}}$, $r_j = 1$, $\beta = \left((5p)^\alpha r^{-p}\Psi_0\right)^{-1/(1+\alpha)}$,

$$A_T^{\frac{1}{1+\alpha}} \geq \exp\Big(\left((5p)^\alpha r^{-p}\Psi_0\right)^{-1/(1+\alpha)} \cdot T - 1\Big).$$

Therefore, when $T = O\left(p^{p/(p+1)}\left(\frac{\Psi_0^{1/p}}{r}\right)^{\frac{p}{p+1}} \log\frac{\Psi_0}{\epsilon}\right)$

$$A_T \geq \frac{\Psi_0}{\epsilon}.$$

Further, using Eq. (8)

$$f(x_T) - f(x^\star) \leq \frac{\Psi_0}{A_T} \leq \epsilon.$$

Additionally noting that $\Psi_0 = \|x^\star - x_0\|_p^p$ we get that in $T = O\left(p^{p/(p+1)}\left(\frac{\|x_0 - x^\star\|_p}{r}\right)^{\frac{p}{p+1}} \log\frac{\Psi_0}{\epsilon}\right)$ iterations, we obtain an ϵ-approximate solution.

■

Appendix C. Proofs for Line-Search Free Method

C.1. Reductions from the σ-*GMS*$_p$ Oracle

We first show a reduction to a general σ-*MS*$_p$ oracle. This is the same as the MS oracles from Monteiro and Svaiter (2013); Carmon et al. (2022) when $p = 2$.

Definition 37 *An oracle $\mathcal{O}_{\sigma,p} : \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}_+$ is a σ-MS$_p$ oracle for a function $f : \mathbb{R}^d \to \mathbb{R}$ if, for every $c \in \mathbb{R}^d$, $(x, \gamma) = \mathcal{O}_{\sigma,p}(c)$ satisfies*

$$\left\|\gamma \textsc{Diag}\left(|x-c|^{p-2}\right)(x-c) + \nabla f(x)\right\|_{p^*} \leq \sigma\gamma\|x-c\|_p^{p-1}.$$

Additionally, we say $\mathcal{O}_{\sigma,p}$ satisfies an (s,μ) movement bound if

$$\|x-c\|_p \geq \begin{cases} (\gamma/\mu^s)^{1/(s-p)} & \text{if } s < \infty \\ 1/\mu & \text{if } s = \infty. \end{cases}$$

We now show how the σ-*MS*$_p$ oracle is related to the σ-*GMS*$_p$ oracle.

Lemma 38 *If $(x,\gamma) = \sigma$-MS$_p(y)$, then*

$$\langle \nabla f(x), x - y\rangle \leq -\gamma(1-\sigma)\|x-y\|_p^p, \text{ and } \|\nabla f(x)\|_{p^*} \leq (1+\sigma)\gamma\|x-y\|_p^{p-1}. \tag{9}$$

Proof Observe that

$$\begin{aligned}
&\langle \nabla f(x), x - y\rangle \\
&= \langle \nabla f(x) + \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-y) - \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-y), x-y\rangle \\
&\leq \langle \nabla f(x) + \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-y), x-y\rangle - \gamma \|x-y\|_p^p \\
&\leq \big\|\nabla f(x) + \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-y)\big\|_{p^*} \|x-y\|_p - \gamma \|x-y\|_p^p \\
&\leq \sigma\gamma \|x-y\|_p^p - \gamma \|x-y\|_p^p,
\end{aligned}$$

as required. The last inequality above follows from the definition of a σ-MS$_p$ oracle. We now show the bound on $\|\nabla f(x)\|_{p^*}$.

$$\begin{aligned}
\|\nabla f(x)\|_{p^*} &= \|\nabla f(x) + \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-c) - \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-c) + \|_{p^*} \\
&\leq \|\nabla f(x) + \gamma \textsc{Diag}\big(|x-c|^{p-2}\big)(x-c)\|_{p^*} + \gamma \|\textsc{Diag}\big(|x-c|^{p-2}\big)(x-c) + \|_{p^*} \\
&\leq \sigma\gamma \|x-c\|_p^{p-1} + \gamma \|x-c\|_p^{p-1},
\end{aligned}$$

where the last step follows from the definition of p^* and the σ-*MS$_p$* oracle. ■

We now prove that the output of an ℓ_p^s oracle is a 0-GMS$_p$ oracle.

Lemma 39 *An oracle that returns (x, γ) with input $c \in \mathbb{R}^d$, $f : \mathbb{R}^d \to \mathbb{R}$, and*

$$x = \text{ the output of a } \ell_p^s(\lambda)\text{-proximal oracle for function } f \text{ queried at } c \text{ and,} \gamma = \begin{cases} \lambda s \|x-c\|_p^{s-p} & \text{if } s < \infty \\ \frac{\|\nabla f(x)\|_{p^*}}{r^{p-1}} & \text{if } s = \infty, \end{cases}$$

is a 0-GMS$_p$ oracle that satisfies a $(s, \min\{1/r, (\lambda s)^{1/s}\})$ movement bound.

Proof Let x be an output of the oracle with input c, f. From the optimality conditions for $s < \infty$,

$$\nabla f(x) = -\lambda s \|x-c\|_p^{s-p} \textsc{Diag}\big(|x-c|^{p-2}\big)(x-c),$$

and for $s = \infty$ (from Lemma 35),

$$\nabla f(x) = -\frac{\|\nabla f(x)\|_{p^*}}{r^{p-1}} \textsc{Diag}\big(|x-c|^{p-2}\big)(x-c).$$

Therefore, in both cases, $\langle \nabla f(x), x-c\rangle = -\gamma \|x-c\|_p^p$. Furthermore, in both cases by a simple computation we see that $\|\nabla f(x)\|_{p*} = \gamma \|x-c\|_p^{p-1}$, as required. We next show the movement bounds. For $s = \infty$, from Lemma 35 $\|x-c\|_p = r$, and for $s < \infty$,

$$\gamma^{\frac{1}{s-p}} \mu^{-\frac{s}{s-p}} = (\lambda s)^{\frac{1}{s-p}} \mu^{-\frac{s}{s-p}} \|x-c\|_p.$$

The bound now follows from requiring $\mu \geq 1/r$ and $\mu \geq (\lambda s)^{1/s}$. ■

With the above result on $\ell_p^s(\lambda)$-oracles in hand, Theorems 4 and 7 follow as corollaries of Theorems 14 and 15 respectively.

C.2. Proof of Lemma 16

Proof Observe,

$$\begin{aligned}
&A'_{t+1}\big(f(x'_{t+1}) - f(x^\star)\big) - A_t(f(x_t) - f(x^\star)) \\
&= a_{t+1}\big(f(x'_{t+1}) - f(x^\star)\big) + A_t\big(f(x'_{t+1}) - f(x_t)\big) \\
&\le a_{t+1}\langle \nabla f(x'_{t+1}), x'_{t+1} - x^\star\rangle + A_t\langle \nabla f(x'_{t+1}), x'_{t+1} - x_t\rangle \quad \text{(convexity of } f\text{)} \\
&= \langle \nabla f(x'_{t+1}), A'_{t+1}x'_{t+1} - A_t x_t - a_{t+1}x^\star\rangle \\
&= \langle \nabla f(x'_{t+1}), A'_{t+1}x'_{t+1} + a_{t+1}z_t - A'_{t+1}y_t - a_{t+1}x^\star\rangle \qquad \text{(Since } A_{t+1}y_t = A_t x_t + a_{t+1}z_t\text{)} \\
&= A'_{t+1}\langle \nabla f(x'_{t+1}), x'_{t+1} - y_t\rangle + a_{t+1}\langle \nabla f(x'_{t+1}), z_t - x^\star\rangle
\end{aligned}$$

Using the KKT conditions for z_t, i.e., differentiating the equation defining z_t in the algorithm w.r.t z, we have

$$\sum_{i=1}^{t} a_i\beta_i\nabla f(x'_i) = -\nabla\omega_p(z_t, y_0), \quad \forall t.$$

Therefore,

$$a_{t+1}\beta_{t+1}\nabla f(x'_{t+1}) = \nabla\omega_p(z_t, y_0) - \nabla\omega_p(z_{t+1}, y_0)$$

This implies

$$\begin{aligned}
a_{t+1}\beta_{t+1}\langle \nabla f(x'_{t+1}), z_{t+1} - x^\star\rangle &= \langle \nabla\omega_p(z_t, y_0) - \nabla\omega_p(z_{t+1}, y_0), z_{t+1} - x^\star\rangle \\
&= \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) - \omega_p(z_t, z_{t+1}) \\
&\le \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) - \frac{1}{2^{p+1}}\|z_{t+1} - z_t\|_p^p
\end{aligned}$$

where the second equality is the Bregman three-point property (Lemma 29) and the inequality uses Lemma 30. We additionally bound

$$a_{t+1}\langle \nabla f(x'_{t+1}), z_{t+1} - z_t\rangle \le a_{t+1}\big\|\nabla f(x'_{t+1})\big\|_{p^*}\|z_{t+1} - z_t\|_p$$

by Hölder's inequality. Combining these inequalities yields

$$\begin{aligned}
a_{t+1}\langle \nabla f(x'_{t+1}), z_t - x^\star\rangle &\le \beta_{t+1}^{-1}(\omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) - \frac{1}{2^{p+1}}\|z_{t+1} - z_t\|_p^p) \\
&\quad + a_{t+1}\big\|\nabla f(x'_{t+1})\big\|_{p^*}\|z_{t+1} - z_t\|_p \\
&\le \beta_{t+1}^{-1}(\omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1})) + \frac{p-1}{p}(2a_{t+1}\beta_{t+1}^{1/p})^{p^*}\big\|\nabla f(x'_{t+1})\big\|_{p^*}^{p^*} \\
&\quad + \frac{1}{\beta_{t+1}p2^p}\|z_{t+1} - z_t\|_p^p - \frac{1}{\beta_{t+1}2^{p+1}}\|z_{t+1} - z_t\|_p^p \\
&\le \beta_{t+1}^{-1}(\omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1})) + (2a_{t+1}\beta_{t+1}^{1/p})^{p^*}\big\|\nabla f(x'_{t+1})\big\|_{p^*}^{p^*}
\end{aligned}$$

where the second inequality uses Young's inequality. Finally, since x'_{t+1} is returned by σ-$GMS_p(y_t)$, we have

$$\langle \nabla f(x'_{t+1}), x'_{t+1} - y_t\rangle \le -(1-\sigma)\lambda_{t+1}\big\|x'_{t+1} - y_t\big\|_p^p \tag{10}$$

and

$$\left\|\nabla f(x'_{t+1})\right\|_{p^*} \leq (1+\sigma)\lambda_{t+1}\left\|x'_{t+1} - y_t\right\|_p^{p-1}. \tag{11}$$

Putting everything together, we obtain

$$\begin{aligned}
&A'_{t+1}\big(f(x'_{t+1}) - f(x^\star)\big) - A_t(f(x_t) - f(x^\star)) \\
&\leq A'_{t+1}\langle \nabla f(x'_{t+1}), x'_{t+1} - y_t\rangle + a_{t+1}\langle \nabla f(x'_{t+1}), z_t - x^\star\rangle \\
&\leq -A'_{t+1}(1-\sigma)\lambda_{t+1}\left\|x'_{t+1} - y_t\right\|_p^p + (2a_{t+1}\beta_{t+1}^{1/p})^{p^*}\left\|\nabla f(x'_{t+1})\right\|_{p^*}^{p^*} + \beta_{t+1}^{-1}(\omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1})) \\
&\leq \left(((2+2\sigma)a_{t+1}\lambda_{t+1}\beta_{t+1}^{1/p})^{p^*} - A'_{t+1}(1-\sigma)\lambda_{t+1}\right)\left\|x'_{t+1} - y_t\right\|_p^p + \beta_{t+1}^{-1}(\omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1})).
\end{aligned}$$

We now consider the cases where $\beta_{t+1} = 1$ and $\beta_{t+1} < 1$ separately. If $\beta_{t+1} = 1$ we must have $\lambda_{t+1} \leq \bar{\lambda}_{t+1}$, $A_{t+1} = A'_{t+1}$, and $x_{t+1} = x'_{t+1}$. The above therefore yields

$$\begin{aligned}
&A_{t+1}(f(x_{t+1}) - f(x^\star)) - A_t(f(x_t) - f(x^\star)) \\
&\leq \lambda_{t+1}\left(((2+2\sigma)a_{t+1})^{p^*}\lambda_{t+1}^{\frac{1}{p-1}} - A_{t+1}(1-\sigma)\right)\|x_{t+1} - y_t\|_p^p + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}).
\end{aligned}$$

Now by the definition of A_{t+1}, we have

$$A_{t+1} = \left(\frac{3(1+\sigma)}{1-\sigma}a_{t+1}\right)^{p^*}\bar{\lambda}_{t+1}^{\frac{1}{p-1}} \geq \frac{1}{1-\sigma}((2+2\sigma)a_{t+1})^{p^*}\lambda_{t+1}^{\frac{1}{p-1}}$$

and so we have

$$A_{t+1}(f(x_{t+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \leq A_t(f(x_t) - f(x^\star)) + \omega_p(x^\star, z_t).$$

By an equivalent argument to Theorem 33, we observe $A_{t+1}^{1/p} \geq A_t^{1/p} + \frac{1}{\frac{3+3\sigma}{1-\sigma}p^{1/p}\bar{\lambda}_{t+1}^{1/p}} \geq A_t^{1/p} + \frac{1}{\frac{6+6\sigma}{1-\sigma}\bar{\lambda}_{t+1}^{1/p}}$ (as $p^{1/p} \leq 2$ for all p), which combined with the above potential bound yields one case of the lemma.

If $\beta_{t+1} < 1$, we have $\lambda_{t+1} \geq \bar{\lambda}_{t+1}$ and $\beta_{t+1}\lambda_{t+1} = \bar{\lambda}_{t+1}$. We observe, again using Eq. (10),

$$\begin{aligned}
&A_{t+1}(f(x_{t+1}) - f(x^\star)) - A_t(f(x_t) - f(x^\star)) \\
&\leq \beta_{t+1}A'_{t+1}\big(f(x'_{t+1}) - f(x^\star)\big) + (1-\beta_{t+1})A_t(f(x_t) - f(x^\star)) - A_t(f(x_t) - f(x^\star)) \\
&= \beta_{t+1}A'_{t+1}\big(f(x'_{t+1}) - f(x^\star)\big) - \beta_{t+1}A_t(f(x_t) - f(x^\star)) \\
&\leq \beta_{t+1}\left(((2+2\sigma)a_{t+1}\lambda_{t+1}\beta_{t+1}^{1/p})^{p^*} - A'_{t+1}(1-\sigma)\lambda_{t+1}\right)\left\|x'_{t+1} - y_t\right\|_p^p + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) \\
&= \left(((2+2\sigma)a_{t+1})^{p^*}\lambda_{t+1}^{p^*}\beta_{t+1}^{p^*} - A'_{t+1}(1-\sigma)\beta_{t+1}\lambda_{t+1}\right)\left\|x'_{t+1} - y_t\right\|_p^p + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1}) \\
&= \left(((2+2\sigma)a_{t+1})^{p^*}\bar{\lambda}_{t+1}^{p^*} - A'_{t+1}(1-\sigma)\bar{\lambda}_{t+1}\right)\left\|x'_{t+1} - y_t\right\|_p^p + \omega_p(x^\star, z_t) - \omega_p(x^\star, z_{t+1})
\end{aligned}$$

where the last equality uses the definition of β_{t+1} and the first inequality uses convexity of f, the definition of x_{t+1}, and that $A_{t+1} = \beta_{t+1}A'_{t+1} + (1-\beta_{t+1})A_t = A_t + \beta_{t+1}a_{t+1}$. The result now follows from $A'_{t+1} = (\frac{3+3\sigma}{1-\sigma}a_{t+1})^{p^*}\bar{\lambda}_{t+1}^{\frac{1}{p-1}}$. ■

C.3. Proof of Lemma 14

Proof We first observe that

$$f(x_{T+1}) - f(x^\star) \le \frac{\Psi_0}{A_{T+1}}.$$

We will thus bound the right-hand side expression. We bound the number of iterations required for A_t to increase by a factor of 2. Let $T_1 \ge 1$ be given and let T_2 be the smallest value where $A_{T_2} \ge 2A_{T_1}$. We will categorize these iterations by whether $\bar\lambda_{t+1} \ge \lambda_{t+1}$ or $\bar\lambda_{t+1} < \lambda_{t+1}$. Let S, L be the sets of iterations where $\bar\lambda_{t+1} \ge \lambda_{t+1}$ and $\bar\lambda_{t+1} < \lambda_{t+1}$ respectively and note that $S \cup L = [T_1, T_2]$.

Summing the bound of Lemma 16 over all iterations and recalling $A_0 = 0$ gives

$$A_{T+1}(f(x_{T+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \le \omega_p(x^\star, x_0) - \sum_{t\in L} \frac{1-\sigma}{3} A'_{t+1}\bar\lambda_{t+1}\|x'_{t+1} - y_t\|_p^p.$$

Since the oracle satisfies the (s,μ) movement bounds for all t, $\|x'_{t+1} - y_t\|_p^p \ge \lambda_{t+1}^{\frac{p}{s-p}}\mu^{-\frac{sp}{s-p}}$. Further since for all $t \in L$, $\lambda_{t+1} > \bar\lambda_{t+1}$, the above becomes,

$$\|x'_{t+1} - y_t\|_p^p \ge \bar\lambda_{t+1}^{\frac{p}{s-p}}\mu^{-\frac{sp}{s-p}}.$$

Since $f(x_T) - f(x^\star) \ge 0$ and $\omega_p(x^\star, z_t) \ge 0$ for all t. This implies that

$$\frac{1-\sigma}{3}\mu^{-\frac{sp}{s-p}} \sum_{t\in L} A'_{t+1}\bar\lambda_{t+1}^{\frac{s}{s-p}} \le \omega_p(x^\star, z_0) = \Psi_0$$

We now observe, by our choice in Algorithm 2, that $\bar\lambda_{t+1} = \bar\lambda = A_{T_1}^{-\frac{(s-p)(p+1)}{ps-p+s}}\Psi_0^{\frac{p(s-p)}{ps-p+s}}\mu^{\frac{sp^2}{s-p+ps}}$ for all $t \in [T_1, T_2]$. Since $A'_{t+1} \ge A_{T_1}$, this implies

$$\frac{1-\sigma}{3}\mu^{-\frac{sp}{s-p}} A_{T_1}\bar\lambda^{\frac{s}{s-p}} \cdot |L| \le \Psi_0,$$

or,

$$|L| \le \frac{3}{1-\sigma}\mu^{\frac{sp}{s-p}}\bar\lambda^{-\frac{s}{s-p}}\frac{\Psi_0}{A_{T_1}}.$$

We will now bound the size of set S. Since for every $t \in S$ we have $A_{t+1}^{1/p} \ge A_t^{1/p} + \frac{1}{\frac{6+6\sigma}{1-\sigma}\bar\lambda_{t+1}^{1/p}} = A_t^{1/p} + \frac{1}{\frac{6+6\sigma}{1-\sigma}\bar\lambda^{1/p}}$, we conclude

$$2^{1/p}A_{T_1}^{1/p} \ge A_{T_2-1}^{1/p} \ge A_{T_1}^{1/p} + \frac{|S|}{\frac{6+6\sigma}{1-\sigma}\bar\lambda^{1/p}} \Rightarrow |S| < \frac{6+6\sigma}{1-\sigma}(A_{T_1}\bar\lambda)^{1/p}.$$

Summing these gives

$$T_2 - T_1 = |S| + |L| \le \frac{6+6\sigma}{1-\sigma}(A_{T_1}\bar\lambda)^{1/p} + \frac{3}{1-\sigma}\mu^{\frac{sp}{s-p}}\frac{\Psi_0}{A_{T_1}\bar\lambda^{\frac{s}{s-p}}}.$$

By our choice of $\bar\lambda$, we have

$$T_2 - T_1 \le \frac{24}{1-\sigma}A_{T_1}^{\frac{p}{ps-p+s}}\Psi_0^{\frac{s-p}{ps-p+s}}\mu^{\frac{sp}{s-p+ps}}.$$

We now sum this bound over all phases encountered in the T iterations. The sum over phases is the sum of a geometric series with initial term $O\left(A_1^{\frac{p}{ps-p+s}}\Psi_0^{\frac{s-p}{ps-p+s}}\mu^{\frac{sp}{s-p+ps}}\right)$ and common ratio $2^{\frac{p}{ps-p+s}}$. We now use the following property of a geometric series with n terms, starting term a and common ratio r:

$$\sum_{i=0}^{n-1} ar^i = a\left(\frac{r^n-1}{r-1}\right) \leq ar^{n-1}\left(\frac{r}{r-1}\right).$$

Note that ar^{n-1} is the last term of the series. Applying this to our series, since $\frac{r}{r-1} = O(1)$, we get,

$$T \leq O\left(A_{T+1}^{\frac{p}{ps-p+s}}\Psi_0^{\frac{s-p}{ps-p+s}}\mu^{\frac{sp}{s-p+ps}}\right).$$

which implies

$$\Psi_0 A_{T+1}^{-1} \leq O\left(\frac{\mu^s\Psi_0^{s/p}}{T^{(ps-p+s)/p}}\right).$$

The result follows with the observation that $\frac{ps-p+s}{p} = s(1+\nu)$ for $\nu = \frac{1}{p} - \frac{1}{s}$.

In addition to the cost of querying the σ-*GMS*$_p$ oracle, the remaining computational costs are bounded by the costs to update y_t, x_{t+1}, and z_{t+1} (which can be expressed in closed-form as a per-coordinate scaling of the (weighted) accumulation of previous gradients), which are $O(d)$. In addition, δ-approximately finding a positive root of the polynomial $m(a) = \bar{\lambda}_{t+1}^{1/p-1}(\frac{3+3\sigma}{1-\sigma}a)^{p/(p-1)} - a - A_t = 0$ (i.e., $\tilde{a}$ such that $|\tilde{a} - a_{\text{root}}| \leq \delta$), is bounded, for $u = (\frac{1-\sigma}{3+3\sigma})\frac{A_t^{(p-1)/p}}{\bar{\lambda}_{t+1}^{1/p}} + \bar{\lambda}_{t+1}^{-(p-1)}(\frac{3+3\sigma}{1-\sigma})^{-p}$, by $O(\log(u/\delta))$, since $m(0) = -A_t \leq 0$ and $m(u) \geq 0$, though we note that this cost is dominated by $O(d)$. ■

C.4. Proof of Lemma 15

Proof We follow the proof of the previous lemma. Define S and L as in the proof of Lemma 14. Again, summing the bound of Theorem 16 over all iterations and recalling $A_0 = 0$ gives

$$A_{T+1}(f(x_{T+1}) - f(x^\star)) + \omega_p(x^\star, z_{t+1}) \leq \omega_p(x^\star, x_0) - \frac{1-\sigma}{3}\sum_{t\in L} A'_{t+1}\bar{\lambda}_{t+1}\|x'_{t+1} - y_t\|_p^p.$$

Using the fact that the oracle satisfies the (∞, μ) movement bounds, we know that $\|x'_{t+1} - y_t\|_p^p \geq 1/\mu^p$ for all t.

Since $f(x_T) - f(x^\star) \geq 0$ and $\omega_p(x^\star, z_t) \geq 0$ for all t. This implies that

$$\frac{1-\sigma}{3}\mu^{-p}\sum_{t\in L} A'_{t+1}\bar{\lambda}_{t+1} \leq \omega_p(x^\star, z_0) = \Psi_0$$

We now choose $\bar{\lambda}_t = \bar{\lambda}$ for all $t \in [T_1, T_2]$. We will choose this value $\bar{\lambda}$ in the end. Since $A'_{t+1} \geq A_{T_1}$, this implies

$$|L| \leq \frac{3}{1-\sigma}\frac{\mu^p\Psi_0}{\bar{\lambda}A_{T_1}}.$$

The bound on $|S|$ is identical to Lemma 14:

$$|S| < \frac{6+6\sigma}{1-\sigma}(A_{T_1}\bar{\lambda})^{1/p}.$$

Summing these gives

$$T_2 - T_1 = |S| + |L| \leq \frac{6+6\sigma}{1-\sigma}(A_{T_1}\bar{\lambda})^{1/p} + \frac{3}{1-\sigma}\frac{\mu^p \Psi_0}{\bar{\lambda} A_{T_1}}.$$

Setting

$$\bar{\lambda} = \frac{\mu^{\frac{p^2}{p+1}} \Psi_0^{\frac{p}{p+1}}}{A_{T_1}},$$

for these iterations yields

$$T = T_2 - T_1 \leq \frac{24}{1-\sigma}\mu^{\frac{p}{p+1}} \Psi_0^{\frac{1}{p+1}}.$$

Now, in every T iterations, the value of A_T doubles. Therefore, we need a total of

$$T = O\left(\mu^{\frac{p}{p+1}} \Psi_0^{\frac{1}{p+1}} \log \frac{\Psi_0}{\epsilon A_0}\right)$$

iterations to get $f(x_T) - f(x^\star) \leq \frac{\Psi_0}{A_T} \leq \epsilon$. ■

Appendix D. Proofs for Applications

D.1. Proof of Theorem 2

Leveraging our framework from Section 3 for the ℓ_s-regression problem, we get as an immediate corollary of Theorem 4,

Corollary 40 *Algorithm 2 applied to $f(x) = \|Ax - b\|_s^s$ finds $\tilde{x}$ such that for all k, and $x^\star := \arg\min f(x)$*

$$f(\tilde{x}) - f(x^\star) \leq \frac{s\|Ax^\star - Ax_0\|_p^s}{k^{\frac{s(p+1)-p}{p}}}$$

Each iteration k involves solving a proximal subproblem of the form,

$$\min_x f(x) + \lambda_k \|A(x - c_k)\|_p^s,$$

for given constant λ_k and vector c_k.

We would use this corollary as the basis to prove our result. We will prove a bound on $\|Ax^\star - Ax_0\|_p^s$ and show how to solve every proximal subproblem efficiently using $\tilde{O}(1)$ smoothed ℓ_p-regression problems. We begin by proving a bound on $\|Ax^\star - Ax_0\|_p^s$.

Lemma 41 *For any $A \in \mathbb{R}^{n\times d}$, $2 \leq p \leq s$, let $f(x) = \|Ax - b\|_s^s$ and $x^\star = \arg\min_x f(x)$. Then,*

$$\|A(x - x^*)\|_p \leq 2^{1+1/s} n^\nu \cdot (f(x) - f(x^*))^{1/s}.$$

Proof Let $x \in \mathbb{R}^d$ be fixed and let $\mathcal{E} := f(x) - f(x^\star)$. We first note from Lemma 30,

$$f(x) - f(x^\star) - \langle \nabla f(x^\star), x - x^\star \rangle \geq \frac{1}{2^{s+1}} \|A(x - x^\star)\|_s^s,$$

and since $\nabla f(x^\star) = 0$,

$$\|A(x - x^\star)\|_s \leq 2^{1+1/s}(f(x) - f(x^\star))^{1/s} = 2^{1+1/s}\mathcal{E}^{1/s}.$$

Using the relation between norms, i.e., for $s \geq p$, and any $z \in \mathbb{R}^n$, $\|z\|_s \leq \|z\|_p \leq n^{\frac{1}{p}-\frac{1}{s}}\|z\|_s$ we can further bound,

$$\|A(x - x^\star)\|_p \leq n^{\frac{1}{p}-\frac{1}{s}} \cdot 2^{1+1/s}\mathcal{E}^{1/s},$$

as required. ■

Solving proximal problems efficiently. We require solving proximal problems of the form,

$$\min_x \|Ax - b\|_s^s + \lambda \|A(x - x_t)\|_p^s$$

We will prove the following.

Lemma 42 *There is an algorithm that can solve the $\ell_p^s(\lambda)$-proximal point problem using $\widetilde{O}(1)$ problems of the form*

$$\min_x d^\top x + \|x - x_t\|_{\nabla^2 f(x_t)}^2 + O(\lambda_t)\|A(x - x_t)\|_p^p$$

We first bound the Hessian of the prox problem.

Lemma 43 *Let $s, p \geq 2$ and define $f(x) = \|Ax - b\|_s^s$. For any y, define $f_y(x) = f(x) + C_s\|A(y - x)\|_p^s$ and $h_y(x) = \|x - y\|_{\nabla^2 f(y)}^2 + C_s\|A(y - x)\|_p^s$. Then for $C_s = e \cdot s^s$, for any x,*

$$\frac{1}{e}\nabla^2 h_y(x) \preceq \nabla^2 f_y(x) \preceq e \cdot \nabla^2 h_y(x).$$

Proof This result follows from a small tweak to the proof of Lemma 4.3 of Jambulapati et al. (2022). We include the proof here for completeness. Observe that,

$$\nabla^2 f(x) = s(s-1)A^\top \textsc{Diag}(|Ax - b|)^{s-2}A.$$

For any vector z,

$$\begin{aligned} z^\top \nabla^2 f(x) z &= s(s-1)\sum_{i\in[n]} |Ax - b|_i^{s-2}(Az)_i^2 \\ &= s(s-1)\sum_{i\in[n]} |Ay - b + A(x - y)|_i^{s-2}(Az)_i^2 \\ &\leq \sum_i \big(es(s-1)|Ay - b|_i^{s-2} + s^s|A(x - y)|_i^{s-2}\big)(Az)_i^2, \end{aligned}$$

where the last line follows from Lemma 4.1 of Jambulapati et al. (2022). We will now use Hölder's inequality and the fact that for all vectors w, $\|w\|_\infty \le \|w\|_p$,

$$\begin{aligned}\sum_i s^s |A(x-y)|_i^{s-2}(Az)_i^2 &= s^s \sum_i |A(x-y)|_i^{s-p}|A(x-y)|_i^{p-2}(Az)_i^2 \\ &\le s^s \|A(x-y)\|_\infty^{s-p} \sum_i |A(x-y)|_i^{p-2}(Az)_i^2 \\ &\le s^s \|A(x-y)\|_p^{s-p}\|z\|^2_{A^\top \textsc{Diag}(|A(x-y)|^{p-2})A}.\end{aligned}$$

Combining yields that,

$$\begin{aligned}z^\top \nabla^2 f(x) z &\le es(s-1) z^\top A^\top \textsc{Diag}(|Ay-b|)^{s-2}Az + s^s\|A(x-y)\|_p^{s-p}\|z\|^2_{A^\top \textsc{Diag}(|A(x-y)|^{p-2})A} \\ &\le e\|z\|_{\nabla^2 f(y)} + s^s\|A(x-y)\|_p^{s-p}\|z\|^2_{A^\top \textsc{Diag}(|A(x-y)|^{p-2})A}. \qquad (12)\end{aligned}$$

Now, let $g_y(x) = C_s\|A(y-x)\|_p^s$. By a simple calculation, we have that

$$\begin{aligned}\nabla^2 g_y(x) =& C_s s(p-1)\|A(x-y)\|_p^{s-p} A^\top \textsc{Diag}(|A(x-y)|)^{p-2}A + C_s s(s-p)\|A(x-y)\|_p^{s-2p} \\ & A^\top \textsc{Diag}(|A(x-y)|)^{p-2}|A(x-y)||A(x-y)|^\top \textsc{Diag}(|A(x-y)|)^{p-2}A \\ \succeq\ & sC_s\|A(x-y)\|_p^{s-p}A^\top \textsc{Diag}(|A(x-y)|)^{p-2}A \qquad (13)\end{aligned}$$

We will now prove the upper and lower bounds on the Hessian of $f_y(x)$. We first show the upper bound $\nabla^2 f_y(x) \preceq e\cdot\nabla^2 h_y(x)$.

$$\begin{aligned}\frac{1}{e}\nabla^2 f_y(x) &= \frac{1}{e}\nabla^2 f(x) + \frac{1}{e}\nabla^2 g_y(x) \\ &\preceq \nabla^2 f(y) + \frac{s^s}{e}\|A(x-y)\|_p^{s-p}A^\top \textsc{Diag}(|A(x-y)|)^{p-2}A + \frac{1}{e}\nabla^2 g_y(x) && \text{(From Eq. (12))} \\ &\preceq \nabla^2 f(y) + \nabla^2 g_y(x) && \text{(From Eq. (13) and } C_s \ge e\cdot s^s) \\ &= \nabla^2 h_y(x).\end{aligned}$$

For the lower bound, switching x and y in (12) gives,

$$z^\top \nabla^2 f(x) z \ge \frac{1}{e}\|z\|_{\nabla^2 f(y)} - \frac{s^s}{e}\|A(x-y)\|_p^{s-2}\|z\|^2_{A^\top A}.$$

As a result,

$$\nabla^2 f_y(x) = \nabla^2 f(x) + \nabla^2 g_y(x) \succeq \frac{1}{e}\cdot\big(\nabla^2 h_y(x) - \nabla^2 g_y(x)\big) - \frac{1}{es}\nabla^2 g_y(x) + \nabla^2 g_y(x) \succeq \frac{1}{e}\cdot\nabla^2 h_y(x).$$

■

Similar to Jambulapati et al. (2022), we use the relative smoothness framework from Lu et al. (2018).

Lemma 44 (Lemma 4.4 (Theorem 3.1 from Lu et al. (2018))) *Let f, h be convex twice-differentiable functions satisfying*

$$\mu\nabla^2 h(x) \preceq \nabla^2 f(x) \preceq L\nabla^2 h(x)$$

for all x. There is an algorithm which given a point x_0 computes a point x with

$$f(x) - \arg\min_y f(y) \leq \epsilon\left(f(x_0) - \arg\min_y f(y)\right)$$

in $O\left(\frac{L}{\mu}\log\frac{1}{\epsilon}\right)$ iterations, where each iteration requires computing gradients of f and h at a point, $O(n)$ additional work, and solving a subproblem of the form

$$\min\{\langle g, x\rangle + Lh(x)\}$$

for vectors g.

We will now prove Lemma 42.

Proof [Proof of Lemma 42] This follows from Lemma 43 and Lemma 44. ■

We now state and prove our main reduction theorem.

Proof [Proof of Theorem 2]

From Corollary 40 after k iterations, Algorithm 1 finds $\tilde{x}$ such that for f denoting the ℓ_s-regression problem,

$$f(\tilde{x}) - f(x^\star) \leq \frac{s\|Ax^\star - Ax_0\|_p^s}{k^{\frac{s(p+1)-p}{p}}}.$$

From Lemma 41, for for some x_0, $\|Ax^\star - Ax_0\|_p^p \leq 2^{p+p/s}n^{\frac{s-p}{s}}(f(x_0) - f(x^\star))^{p/s}$. Therefore,

$$f(\tilde{x}) - f(x^\star) \leq \frac{s2^{s+1}n^{\frac{s-p}{p}}}{k^{\frac{s(p+1)-p}{p}}}(f(x) - f(x^\star)).$$

The above implies that in $k = O\left(n^{\frac{s-p}{s(p+1)-p}}\right)$ iterations, the function error reduces by $1/2$ and as a result in at most $O\left(n^{\frac{s-p}{s(p+1)-p}}\log\frac{f(x_0)-f(x^\star)}{\epsilon f(x^\star)}\right)$ iterations, the function error reduces to $\epsilon f(x^\star)$. In other words, we get an ϵ-approximation to the ℓ_s-regression problem.

Furthermore, from the relations between s and p norms, we can find a starting solution that is an $O\left(n^{s\left(\frac{1}{p}-\frac{1}{s}\right)}\right)$-approximate solution to the ℓ_s-regression problem by solving one problem of the form of Eq (2). So in at most

$$O\left(sn^{\frac{s-p}{s(p+1)-p}}\log\frac{n}{\epsilon}\right)$$

iterations, each iteration solving a prox problem, we can find an ϵ-approximation to the ℓ_s-regression problem.

Furthermore, from Lemma 42, the proximal subproblem required to solve in each iteration can be reduced to solving $\widetilde{O}(1)$ problems of the form,

$$\min_x d^\top x + \|x - c\|_{\nabla^2 f(c)}^2 + \lambda_t\|x - c\|_p^p.$$

This concludes the proof of our result. ■

D.2. Proof of Theorem 20

Proof We have

$$\begin{aligned}\langle \nabla f(\tilde{x}),\, \tilde{x} - y\rangle &= \langle \nabla f(\tilde{x}),\, \tilde{x} - x^*_{c,r}\rangle + \langle \nabla f(\tilde{x}) - \nabla f(x^*_{c,r}),\, x^*_{c,r} - y\rangle + \langle \nabla f(x^*_{c,r}),\, x^*_{c,r} - y\rangle \\ &\le \|\nabla f(\tilde{x})\|_{p^*}\|\tilde{x} - x^*_{c,r}\|_p + \|\nabla f(\tilde{x}) - \nabla f(x^*_{c,r})\|_{p^*}\|x^*_{c,r} - y\|_p + \langle \nabla f(x^*_{c,r}),\, x^*_{c,r} - y\rangle,\end{aligned}$$

where we use Holder's inequality. Now, since $\tilde{x}$ is the output of $\ell_p^\infty(\delta, r)$ we have $\|\tilde{x} - x^*_{c,r}\|_p \le \delta$ and $\|x^*_{c,r} - c\|_p \le r$. Furthermore, since f is smooth,

$$\|\nabla f(\tilde{x}) - \nabla f(x^*_{c,r})\|_{p^*} \le L\|\tilde{x} - x^*_{c,r}\|_p \le L\delta.$$

Using these in the above gives,

$$\langle \nabla f(\tilde{x}),\, \tilde{x} - y\rangle \le \delta\|\nabla f(\tilde{x})\|_{p^*} + Lr\delta + \langle \nabla f(x^*_{c,r}),\, x^*_{c,r} - y\rangle.$$

Since $x^*_{c,r}$ is the solution of a ball oracle, from Lemma 35, $\|x^*_{c,r} - c\|_p = r$ and

$$\nabla f(x^*_{c,r}) = -\frac{\|\nabla f(x^*_{c,r})\|_{p/(p-1)}}{r^{p-1}}\textsc{Diag}\big(|x^*_{c,r} - c|\big)^{p-2}(x^*_{c,r} - c).$$

As a result,

$$\begin{aligned}\langle \nabla f(\tilde{x}),\, \tilde{x} - y\rangle &\le \delta\|\nabla f(\tilde{x})\|_{p^*} + Lr\delta - \frac{\|\nabla f(x^*_{c,r})\|_{p^*}}{r^{p-1}}\|x^*_{c,r} - y\|_p^p \\ &= \delta\|\nabla f(\tilde{x})\|_{p^*} + L\delta r - \|\nabla f(x^*_{c,r})\|_{p^*}\|x^*_{c,r} - y\|_p.\end{aligned}$$

Now, using the triangle inequality,

$$\begin{aligned}\langle \nabla f(\tilde{x}),\, \tilde{x} - y\rangle &\le \delta\|\nabla f(\tilde{x})\|_{p^*} + L\delta r - \|\nabla f(x^*_{c,r})\|_{p^*}\|\tilde{x} - y\|_p \\ &\quad + \|\nabla f(x^*_{c,r})\|_{p^*}\|x^*_{c,r} - \tilde{x}\|_p \\ &\le \delta\|\nabla f(\tilde{x})\|_{p^*} + L\delta r - \|\nabla f(x^*_{c,r})\|_{p^*}\|\tilde{x} - y\|_p \\ &\quad + \|\nabla f(x^*_{c,r}) - \nabla f(\tilde{x})\|_{p^*}\|x^*_{c,r} - \tilde{x}\|_p + \|\nabla f(\tilde{x})\|_{p^*}\|x^*_{c,r} - \tilde{x}\|_p && \text{(Triangle inequality)} \\ &\le 2\delta\|\nabla f(\tilde{x})\|_{p^*} + L\delta(r+\delta) - \|\nabla f(x^*_{c,r})\|_{p^*}\|\tilde{x} - y\|_p && (\|\tilde{x} - x^*_{c,r}\|_p \le \delta, f \text{ is } L \text{ smooth}) \\ &\le 2\delta\|\nabla f(\tilde{x})\|_{p^*} + L\delta(r+\delta) - \|\nabla f(\tilde{x})\|_{p^*}\|\tilde{x} - y\|_p \\ &\quad + \|\nabla f(\tilde{x}) - \nabla f(x^*_{c,r})\|_{p^*}\|\tilde{x} - y\|_p && \text{(Triangle inequality)} \\ &\le 2\delta\|\nabla f(\tilde{x})\|_{p^*} + 2L\delta(r+\delta) - \|\nabla f(\tilde{x})\|_{p^*}\|\tilde{x} - y\|_p \\ &\le -(1-\sigma)\|\nabla f(\tilde{x})\|_{p^*}\|\tilde{x} - y\|_p && \text{(From our choice of } \delta) \\ &= -(1-\sigma)\frac{\|\nabla f(\tilde{x})\|_{p^*}}{\|\tilde{x} - y\|_p^{p-1}}\|\tilde{x} - y\|_p^p \\ &= -(1-\sigma)\tilde{\gamma}\|\tilde{x} - y\|_p^p.\end{aligned}$$

In addition, we have

$$\|\nabla f(\tilde{x})\|_{p^*} = \frac{\|\nabla f(\tilde{x})\|_{p^*}}{\|\tilde{x} - y\|_p^{p-1}}\|\tilde{x} - y\|_p^{p-1} = \tilde{\gamma}\|\tilde{x} - y\|_p^{p-1} \le (1+\sigma)\tilde{\gamma}\|\tilde{x} - y\|_p^{p-1},$$

and so it follows that the oracle implements a σ-GMS$_p$ oracle. We furthermore may observe that the oracle satisfies an $(\infty, (r-\delta)^{-1})$ movement bound, since

$$\|\tilde{x} - y\| \geq \left\|x^*_{c,r} - y\right\| - \left\|x^*_{c,r} - \tilde{x}\right\| \geq r - \delta.$$

■

D.3. Proof of Theorem 21

Proof The proof follows along similar lines as that for Theorem 10, though slightly modified to account for the smoothness in $\|\cdot\|_{\nabla^2 f(c)}$, as well as the fact that the x_{t+1} update is now constrained. In particular we have that, for all t,

$$\begin{aligned}
\langle \nabla f(x_{t+1}),\, x_{t+1} - y_t \rangle &= \langle \nabla f(x_{t+1}) - \nabla f(x_t),\, x_{t+1} - y_t \rangle + \langle \nabla f(x_t),\, x_{t+1} - y_t \rangle \\
&\leq \|\nabla f(x_{t+1}) - \nabla f(x_t)\|_{(\nabla^2 f(c))^{-1}} \|x_{t+1} - y_t\|_{\nabla^2 f(c)} + \langle \nabla f(y_t),\, x_{t+1} - y_t \rangle \\
&\leq \|\nabla f(x_{t+1}) - \nabla f(x_t)\|_{(\nabla^2 f(c))^{-1}} \|x_{t+1} - y_t\|_{\nabla^2 f(c)} - 2\xi \|x_{t+1} - y_t\|^2_{\nabla^2 f(c)} \\
&\leq -\xi \|x_{t+1} - y_t\|^2_{\nabla^2 f(c)},
\end{aligned}$$

where the second inequality follows from the optimality conditions for x_{t+1}, and the final inequality follows from the smoothness in $\|\cdot\|_{\nabla^2 f(c)}$. We additionally note that, for $d(x) = \|x\|^2_{\nabla^2 f(c)}$, we have $\omega(x, y) = \|x - y\|^2_{\nabla^2 f(c)}$, thus yielding the remaining parts of the proof, so that, letting $x_{k,0}$ and $x_{k,T}$ denote the starting and final iterates of the k^{th} call to Algorithm 3 (i.e., at the k^{th} iteration of Algorithm 4), we arrive at

$$f(x_{k,T}) - f(x^*_{c,r}) \leq \frac{4\xi \left\|x_{k,0} - x^*_{c,r}\right\|^2_{\nabla^2 f(c)}}{T^2}.$$

Finally, the fact that, by Hessian stability, the function is ξ^{-1} strongly convex with respect to $\|\cdot\|_{\nabla^2 f(c)}$, when combined with a standard restarting argument, leads to

$$\left\|x_{K,T} - x^*_{c,r}\right\|^2_p \leq \frac{d^{1-\frac{2}{p}}}{\mu} \|x_K - x^*\|^2_{\nabla^2 f(c)} \leq \frac{\xi d^{1-\frac{2}{p}}}{\mu} (f(x_K) - f(x^*_{c,r})) \leq \delta^2$$

after $KT = O(\xi \log(\frac{\xi d(f(x_0) - f(x^*_{c,r}))}{\mu\delta})) \leq O(\xi \log(\frac{\xi dLr}{\mu\delta}))$ iterations, and so it follows that $\left\|x_{K,T} - x^*_{c,r}\right\|_p \leq \delta$, as desired. ■

Appendix E. Proofs for Lower Bounds

E.1. Useful Lemmas

We begin by observing useful properties of this function and its optimizer $x^\star$, whereby we find it helpful to define, for all $k' \leq k$,

$$S_{k'} := \{x : \text{supp}\{x\} \subseteq \{1, \ldots, k'\}\}.$$

Lemma 45 *Let $k' \leq k$. Then, for any $x \in S_{k'}$,*

$$g_k(x) \geq -(k'+1) \cdot \beta_k \alpha_k.$$

Proof Since $x \in S_{k'}$, we know that $x_{k'+1} = 0$. Now,

$$g_k(x) \geq \beta_k \max_{1 \leq i \leq k} \{x_i - \alpha_k \cdot i\} \geq \beta_k (x_{k'+1} - \alpha_k \cdot (k'+1)) = -\beta_k \alpha_k \cdot (k'+1).$$

■

Lemma 46 *Let $x^\star$ denote the optimizer of Problem (4). Then for $1 \leq k' \leq k$,*

$$g_k(x^\star) \leq -\frac{\beta_k R}{k'^{1/p}}.$$

In addition, we have that $\|x^\star\| \leq 2R$.

Proof Consider the point $y_i = \begin{cases} -R/k'^{1/p} & \text{if } i \leq k' \\ 0 & \text{otherwise.} \end{cases}$ Now, $\|y\|_p \leq R$, and so it follows that

$$\begin{aligned} g_k(x^\star) \leq g_k(y) = \max\Big\{ f_k(y), \beta_k(\|y\|_p - 2R) - \alpha_k \Big\} &\leq \max\Big\{ \frac{-\beta_k R}{k'^{1/p}} - \alpha_k, -\beta_k R - \alpha_k \Big\} \\ &\leq -\frac{\beta_k R}{k'^{1/p}}. \end{aligned}$$

In addition, we may observe that, for any x s.t. $\|x\|_p > 2R$, $g_k(x) \geq \beta_k(\|x\|_p - 2R) - \alpha_k > -\alpha_k$, whereas $g_k(x^\star) \leq -\alpha_k$, meaning we have $\|x^\star\|_p \leq 2R$. ■

We now show that if there is a sequence of iterates $x^{(i)}$, for $1 \leq i \leq k'$, such that $x^{(i)} \in S_i$, then we can give a lower bound.

Lemma 47 *Let $x^{(i)}$, for $1 \leq i \leq k'$ be such that $x^{(i)} \in S_i$ for all i. Then,*

$$g_k(x^{(i)}) - g_k(x^\star) \geq \frac{\beta_k R}{(i+1)^{1/p}} - (i+1) \cdot \beta_k \alpha_k,$$

and so, after k iterations, we have

$$g_k(x_k) - g_k(x^\star) \geq \frac{\lambda R^s}{16(k+1)^{\frac{s(p+1)-p}{p}}}.$$

Proof The proof follows from Lemma 45 and Lemma 46 and using $\left(1 - \frac{1}{s}\right)^{s-1} \geq 1/4$. ■

E.2. Proof of Lemma 25

Lemma 48 *Let $k, d \in \mathbb{N}$ be such that $1 \leq k \leq d$, and let g_k be as in (4). Then, g_k is an $\ell_p^s(\lambda)$-proximal zero-chain.*

Proof We will prove this by induction.

Base case: Let $x = 0$, meaning that $\mathrm{supp}\{x\} = \emptyset$. Now, letting $x' = \mathcal{O}_{g_k,\lambda,p,s}(x) = \mathcal{O}_{g_k,\lambda,p,s}(0)$, we must have from optimality conditions one of the following two cases, depending on which function maximizes $g_k(x)$. (To simplify presentation, we let, $|\boldsymbol{v}|$, $\mathrm{sgn}(\boldsymbol{v})$ denote the coordinate-wise absolute value and sign, respectively, for a vector $\boldsymbol{v}$.)

- Case 1:
$$\left\|x^{(i)}\right\|^{1-p}(x^{(i)})^{p-1} = -\lambda \cdot s\|x^{(i)} - c\|_p^{s-p}|x^{(i)} - c|^{p-1} \cdot \mathrm{sgn}(x^{(i)} - c),$$
where $(x^{(i)})^{p-1} := [(x_1^{(i)})^{p-1}, \ldots, (x_d^{(i)})^{p-1}]^\top$.
- Case 2:
$$\nabla f_k(x^{(i)}) = -\lambda \cdot s\|x^{(i)} - c\|_p^{s-p}|x^{(i)} - c|^{p-1} \cdot \mathrm{sgn}(x^{(i)} - c).$$

For Case 1, it follows that $x' = 0 \in S_1$, and so the zero-respecting condition is satisfied. It remains to handle Case 2, which occurs when the maximizing function is $f_k(x)$. Now, let $x' = -\left(\frac{\beta_k}{s\lambda}\right)^{\frac{1}{s-1}} \boldsymbol{e}_1$. We claim that x' satisfies the optimality conditions. First, from the definition of g_k, we have that
$$g_k(x') = \beta_k \max_{1 \le i \le k} \left(x'_i - i \cdot \alpha_k\right).$$
Now, from the value of α_k, since $x'_1 - \alpha_k > -2\alpha_k > -3\alpha_k, \cdots$, we must have that $g_k(x') = \beta_k(x_1 - \alpha_k)$. Therefore, $\nabla g_k(x') = \beta_k \boldsymbol{e}_1$. Thus, one may verify that the optimality conditions are satisfied by x', which proves the base case since $\mathrm{supp}\{x'\} = \{1\}$.

If $x^{(j)} \in S_j, \forall j \le i-1$: Since $x^{(i)} = \mathcal{O}_{g_k,\lambda,p,s}(c)$, $c \in S_{i-1}$, we must again have from optimality conditions one of the following two cases, depending on which function maximizes $g_k(x)$.

- Case 1:
$$\left\|x^{(i)}\right\|^{1-p}(x^{(i)})^{p-1} = -\lambda \cdot s\|x^{(i)} - c\|_p^{s-p}|x^{(i)} - c|^{p-1} \cdot \mathrm{sgn}(x^{(i)} - c),$$
where $(x^{(i)})^{p-1} := [(x_1^{(i)})^{p-1}, \ldots, (x_d^{(i)})^{p-1}]^\top$.
- Case 2:
$$\nabla f_k(x^{(i)}) = -\lambda \cdot s\|x^{(i)} - c\|_p^{s-p}|x^{(i)} - c|^{p-1} \cdot \mathrm{sgn}(x^{(i)} - c).$$

For Case 1, it follows that, since $c \in S_{i-1}$, we have that $x^{(i)} \in S_{i-1} \subseteq S_i$, and so the zero-respecting condition is satisfied.

It remains to handle Case 2, which occurs when the maximizing function is $f_k(x)$. To begin, consider $x^{(i)} = c - \gamma \boldsymbol{e}_i$ for $\gamma = \left(\frac{\beta_k}{s\lambda}\right)^{1/(s-1)}$. Since, by our choice of γ (as well as α_k), we have that $-\gamma - i\alpha_k > -(i+1)\alpha_k > -(i+2)\alpha_k \cdots$, it follows that $\arg\max_j \beta_k(x_j^{(i)} - j\alpha) \subseteq \{1, \cdots, i\}$. Therefore, we have that $\nabla g_k(x^{(i)}) \in \beta_k \mathrm{conv}(\boldsymbol{e}_j, j \le i)$. Furthermore, we observe that
$$-\lambda \cdot s\|x^{(i)} - c\|_p^{s-p}|x^{(i)} - c|^{p-1} \cdot \mathrm{sgn}(x^{(i)} - c) = \beta \boldsymbol{e}_i \in \beta_k \mathrm{conv}(\boldsymbol{e}_j, j \le i),$$
which establishes that $x^{(i)}$ satisfies the optimality conditions. It follows that $\mathrm{supp}\{x^{(i)}\} \subseteq \{1, 2, \cdots, i\}$.
■

E.3. Proof of Theorem 5

Theorem 5 ($\ell_p^s(\lambda)$**-Proximal Oracle Optimization Lower Bound)** *For every $\ell_p^s(\lambda)$-proximal zero-respecting algorithm (Definition 22) given $p, s \geq 2$, $\lambda > 0$, $\epsilon > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\lambda \|x^\star\|_p^s/\epsilon)^{\frac{1}{s(1+\nu)}})$ iterations of the algorithm satisfies $f(x) - f(x^\star) \geq \epsilon$.*

Proof Let $\mathcal{A}$ be any $\ell_p^s(\lambda)$-proximal zero-respecting algorithm and construct $f = g_k$ as in (4). Now, consider the optimization problem $\min_{\|x\|_p \leq R} f(x)$. From Lemma 25, the first k queries $x^{(i)}, i \leq k$ from $\mathcal{A}$, satisfy $x^{(i)} \in S_i$ for all $i \leq k$. Now, from Lemma 47 for all $i \leq k$,

$$f(x^{(i)}) - f(x^\star) \geq \frac{\lambda R^s}{16(i+1)^{\frac{s(p+1)-p}{p}}}.$$

Thus, for all $i \leq k = O\big(\frac{\lambda R^s}{\epsilon}\big)^{\frac{1}{s(1+\nu)}}$, $f(x^{(i)}) - f(x^\star) \geq \epsilon$. ■

E.4. Proof of Lemma 49

Lemma 49 *Let $k, d \in \mathbb{N}$ be such that $1 \leq k \leq d$, and let g_k be as in (14). Then, g_k is an $\ell_p^\infty(r)$-proximal zero-chain.*

Proof We prove by induction.

Base case: We know that $x^{(0)} = 0$. Now, since $x^{(1)} = \mathcal{O}_{g_k,r,p,\infty}(0)$, we must have $\|x^{(1)}\|_p \leq r$ and using Lagrange duality, we want to solve,

$$\min_x \max_\mu g_1(x) + \mu(\|x\|_p^p - r^p).$$

When $\mu > 0$, we must have $\|x^{(1)}\|_p = r$. If $\mu = 0$ then $\nabla g_k(x^{(1)}) = 0$ and $\|x^{(1)}\|_p < r$. The later case cannot happen since $\nabla g_k(x)$ is not 0 for any x. Therefore, $x^{(1)} = -r\boldsymbol{e}_1$ satisfies the above optimality conditions if the corresponding μ is positive. If that is true, then since the problem is strictly convex $x^{(1)}$ must be the unique solution.

We now verify that for this value of $x^{(1)}$, $\mu > 0$. This is because, $g_k(x^{(1)}) = x^{(1)} - \alpha$ since $\alpha \geq 4r$. As a result $\nabla g_k(x^{(1)}) = \boldsymbol{e}_1$. From the optimality conditions, $\nabla g(x^{(1)}) = -\mu p |x^{(1)}|^{p-2} x^{(1)}$ which gives, $\mu = 1/(pr^{p-1}) > 0$.

If $x^{(j)} \in S_j, \forall j \leq i-1$: Since $x^i = \mathcal{O}_{g_k,r,p,\infty}(c)$, we must have that $\|x^{(i)} - c\|_p = r$ and following a similar reasoning with respect to the optimality conditions as before, it again follows that $x^{(i)} = c - r\boldsymbol{e}_i \in S_i$. ■

E.5. Proof of Theorem 8

Similar to the $s < \infty$ setting, and letting $f_k(x) = \max_{1 \leq i \leq k}\{x_i - i \cdot 4r\}$, we consider the following hard function:

$$g_k(x) := \max\Big\{f_k(x), \|x\|_p - 2R - 4r\Big\}. \tag{14}$$

Now from Lemma 47, for $x^{(i)} \in S_i$ $(i \geq 1)$,

$$g_k(x^{(k)}) - g_k(x^\star) \geq \frac{R}{(i+1)^{1/p}} - 4r(i+1). \tag{15}$$

We can now complete the proof of Theorem 8, which we restate here for convenience.

Theorem 8 (ℓ_p Ball-constrained Oracle Optimization Lower Bound) *For every $\ell_p^\infty(r)$-proximal zero-respecting algorithm (Definition 26) given $p \geq 2$, $r > 0$, there is a function $f : \mathbb{R}^d \to \mathbb{R}$ such that any x in the first $k = O((\|x^\star\|_p/r)^{\frac{p}{p+1}})$ iterations of the algorithm satisfies*

$$f(x) - f(x^\star) \geq \Omega(\|x^\star\|_p^{1/(p+1)} r^{p/(p+1)}) \,.$$

Proof From Lemma 49, $x^{(i)} \in S_i$. Further, from Eq. (15),

$$g_k(x^{(i)}) - g(x^\star) \geq \frac{R}{(i+1)^{1/p}} - 4r(i+1).$$

So for the first $k = O((R/r)^{\frac{p}{p+1}})$ iterations, $g_k(x^{(i)}) - g(x^\star) \geq \Omega(R^{1/(p+1)} r^{p/(p+1)})$. ■