OPTIMAL FIRST-ORDER METHODS FOR A CLASS OF STRUCTURED HÖLDER-SMOOTH PROBLEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Convex optimization with Hölder-continuous gradients has attracted considerable attention as a framework bridging smooth and nonsmooth optimization. For this class, Nesterov's universal fast gradient method attains the optimal black-box complexity, and we study whether exploiting further problem structure can further improve this bound. In this paper, we study an important class of Hölder-smooth problems, namely regularized ℓ_p regression with $p \in (1, 2)$, whose loss has a $(p-1)$-Hölder-continuous gradient and for which the universal method requires $\mathcal{O}(\varepsilon^{-2/(3p-2)})$ iterations. Exploiting the problem structure, we characterize a family of smooth approximations based on p-Huber smoothing and prove that it attains the exact optimal one-sided approximation–smoothness trade-off. Combined with accelerated proximal gradients, the smoothing approach attains an ε-optimal solution in $\mathcal{O}(\varepsilon^{-1/p})$ matrix–vector products, which significantly improves the complexity guarantee for $1 < p < 2$. We show that this bound is optimal by providing a matching lower bound. Numerical experiments on synthetic and real data demonstrate the practical performance of the proposed methods.

1 INTRODUCTION

Consider the following composite ℓ_p regression problem for $p \in [1, 2)$:

$$\min_{\boldsymbol{u} \in \mathbb{R}^n} \left\{ F(\boldsymbol{u}) := f(\boldsymbol{u}) + \omega(\boldsymbol{u}) = \frac{1}{p} \|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_p^p + \omega(\boldsymbol{u}) \right\}, \quad \boldsymbol{A} \in \mathbb{R}^{m \times n}, \boldsymbol{b} \in \mathbb{R}^m. \tag{1}$$

Here, ω represents additional regularization or constraints, such as sparsity-promoting ℓ_1 penalties or box constraints. The range $p \in (1, 2)$ interpolates between least absolute deviations and least squares, and such ℓ_p regression has long been studied as an alternative to least squares under non-Gaussian errors (Money et al., 1982). Related objectives also arise in semi-supervised learning and local graph clustering (Liu & Gleich, 2020; El Alaoui et al., 2016).

For $1 < p < 2$, the gradient of f is $(p-1)$-Hölder continuous. Nesterov's universal fast gradient method attains an ε-optimal solution in $\mathcal{O}(\varepsilon^{-2/(3p-2)})$ iterations without knowing the Hölder exponent or constant in advance, and this complexity is optimal for a generic objective with a Hölder continuous gradient in the standard first-order model (Nesterov, 2015). Thus, to obtain a faster rate for ℓ_p regression, one must exploit the structure of f. One classic example is the endpoint $p = 1$, where the first-order complexity is $\mathcal{O}(\varepsilon^{-2})$, while smoothing the absolute-value loss attains an $\mathcal{O}(\varepsilon^{-1})$ iteration bound (Nesterov, 2005; Beck & Teboulle, 2012). Thus, a natural question is: can the structure of ℓ_p regression be used to improve the general Hölder-class complexity bound for $1 < p < 2$?

A natural starting point is the Moreau envelope of the scalar loss $r \mapsto |r|^p/p$, obtained through a quadratic proximal penalty (Beck & Teboulle, 2012). At $p = 1$, this gives the classical Huber approximation with a uniformly bounded error. For $1 < p < 2$, however, the gap between this scalar loss and its Moreau envelope can be arbitrarily large for large residuals, for any given positive smoothing parameter (Proposition 2.2). We adopt the quadratic smoothing construction of Bubeck et al. (2018), which we call p-Huber smoothing (Definition 2.1). It replaces the loss near zero by a quadratic and preserves the original gradient outside that neighborhood, so the approximation error is uniformly bounded.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Our contributions are as follows. First, we establish an optimality property of the p-Huber smoothing construction: for a prescribed smoothness level, it minimizes the uniform approximation error among smooth lower approximations of the scalar loss $|r|^p/p$. We also give a dual interpretation of this property through its conjugate representation.

Second, we analyze accelerated proximal gradient (APG) with p-Huber smoothing for proximable convex regularizers and $1 \leq p < 2$, including fixed, variable, and backtracking variants. Fixed smoothing attains an ε-optimal solution in $\mathcal{O}(\varepsilon^{-1/p})$ matrix–vector products. We also establish a lower complexity bound for deterministic methods in our oracle model. This bound matches our upper bound up to constants depending only on p.

Finally, we evaluate our method on a broad collection of problems, including synthetic regression, UCI data, sparse text regression, and box-constrained image deblurring, and it is competitive with other first-order baselines. On synthetic data with a small exponent p and an ill-conditioned matrix, our model reaches a lower error than the first-order baselines, and it meets a relative accuracy of 10^{-6} on 201 of the 250 instances, more than any other method in the comparison. Across the UCI problems, it meets the accuracy targets at least as often as the baselines, and one variant uses fewer matrix–vector products. In the text experiments, our variants produce the smallest final error, with one of them ahead when the penalty is large. Our model also reaches the target accuracy with fewer matrix–vector products in image deblurring tasks.

1.1 Related work

First-order methods for Hölder-smooth optimization. For $p \in (1, 2)$, the loss f has a $(p-1)$-Hölder-continuous gradient. Nesterov's universal accelerated method attains the optimal rate for this generic Hölder class and serves as our canonical black-box baseline (Nesterov, 2015). More recent distance-adaptive methods achieve optimal anytime Hölder rates without prior knowledge of the smoothness parameters or target accuracy; we compare with the deterministic unconstrained AGDA method of Ren et al. (2025). The closest structure-aware first-order baseline is Generalized-AGD+ of Diakonikolas & Guzmán (2024), whose general-norm framework contains a direct specialization to powered ℓ_p regression. Its guarantee is expressed in the native ℓ_p geometry, whereas ours uses the Euclidean operator norm and initial distance, so the two parameterizations do not uniformly dominate one another.

Smoothing and alternative geometries. Our analysis follows the classical smoothing-and-acceleration paradigm of Nesterov (2005) and Beck & Teboulle (2012). Homotopy methods already minimize a quadratically smoothed ℓ_p objective (Bubeck et al., 2018). Iterative refinement reduces ℓ_p regression to a short sequence of residual subproblems (Adil et al., 2019a), and algorithms for p-norm flows use smoothed q-norm problems as subroutines (Adil & Sachdeva, 2020). Huber-like and piecewise-power losses have additionally been studied from a sparsification perspective (Jambulapati et al., 2024). A different approach to non-Lipschitz gradients is to change the optimization geometry through relative smoothness and Bregman methods (Bauschke et al., 2017; Lu et al., 2018); in particular, Maddison et al. (2021) develop nonlinear dual-space preconditioning for a complementary $p \geq 2$ regression regime.

Other high-accuracy methods for standard ℓ_p regression. A substantial literature develops IRLS, homotopy, and iterative-refinement methods with strong complexity guarantees for standard ℓ_p regression (Li, 1993; Bubeck et al., 2018; Adil et al., 2019a;b; 2024; Jambulapati et al., 2022). Related advances use Lewis weights and sparsification (Jambulapati et al., 2022; 2024); Ene et al. (2026) develop a primal–dual IRLS method and bound the number of linear system solves it requires. They also extend the method to $1 < p < 2$ through duality. The quadratically smoothed objectives of Bubeck et al. (2018) and mixed quadratic–ℓ_p residual models in iterative refinement are also closely related to the p-Huber smoothing used here. These guarantees are typically formulated for unregularized problems and rely on repeated weighted least-squares or linear-system solves. This computational regime is distinct from the composite matrix–vector oracle model considered here.

2 PROBLEM FORMULATION AND p-HUBER SMOOTHING

We consider the composite ℓ_p regression problem in (1), with $1 \le p < 2$, where ω is a convex regularizer. We assume that $\omega : \mathbb{R}^n \to (-\infty, \infty]$ is proper and lower semicontinuous, with an exactly computable proximal operator $\mathrm{prox}_{\omega/\gamma}$ for every $\gamma > 0$. This includes ℓ_1 penalties and indicator functions of simple convex sets. Throughout, $\|\cdot\|$ denotes the Euclidean norm on vectors and the spectral norm on matrices. We assume that the solution set $\mathcal{U}^\star := \arg\min_{\boldsymbol{u}\in\mathbb{R}^n} F(\boldsymbol{u})$ is nonempty and write $F^\star := \min_{\boldsymbol{u}\in\mathbb{R}^n} F(\boldsymbol{u})$. We also assume that $\|\boldsymbol{A}\| > 0$. Given an initial point $\boldsymbol{u}_0 \in \mathrm{dom}(\omega)$, the algorithm is supplied with a bound $D > 0$ satisfying $D \ge \mathrm{dist}(\boldsymbol{u}_0, \mathcal{U}^\star)$.

The computational model restricts access to $\boldsymbol{A}$ to matrix–vector products $\boldsymbol{A}\boldsymbol{u}$ and $\boldsymbol{A}^\top\boldsymbol{v}$, and provides access to ω through its proximal operator. Matrix factorizations and linear-system solves are not treated as unit-cost primitives. Individual matrix–vector products and proximal evaluations are counted separately. The full deterministic oracle model is specified in Subsection 3.3.

2.1 OPTIMAL SCALAR SMOOTHING

The goal is to construct a smoothing surrogate for the composite objective $F(\boldsymbol{u}) = f(\boldsymbol{u}) + \omega(\boldsymbol{u})$ in (1). We smooth only f and leave ω unchanged. Since f is separable in the residual coordinates $r_i = (\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b})_i$, it is reasonable to first study the scalar loss $\psi(r) := |r|^p/p$.

For $L > 0$, define the class of differentiable L-smooth minorants of ψ as

$$\Phi_L := \{\phi \in \mathcal{C}^1(\mathbb{R}) : \mathrm{Lip}(\phi') \le L, \phi \le \psi\}.$$

For $1 < p < 2$, the curvature $\psi''(r) = (p-1)|r|^{p-2}$ is unbounded near zero but decreases as $|r|$ increases. This suggests replacing the loss near zero by a quadratic function while preserving its derivative outside this region. Following the smoothing constructions of Adil et al. (2019a; 2024); Bubeck et al. (2018), we consider the following p-Huber smoothing of ψ.

Definition 2.1 (p-Huber smoothing)**.** For $1 \le p < 2$ and a smoothing radius $\rho > 0$, the p-Huber smoothing surrogate ϕ_ρ of $\psi(r) = |r|^p/p$ is defined as

$$\phi_\rho(r) := \begin{cases} \frac{1}{2}\rho^{p-2}r^2 & |r| \le \rho, \\ \frac{1}{p}|r|^p - c_p\rho^p & |r| > \rho, \end{cases} \qquad c_p := (2-p)/(2p). \tag{2}$$

For $\rho > 0$, ϕ_ρ is convex and continuously differentiable ($\mathcal{C}^1(\mathbb{R})$). Its derivative $\phi'_\rho(r) = (\max\{\rho, |r|\})^{p-2}r$ has Lipschitz constant ρ^{p-2}, and the approximation error satisfies $0 \le \psi - \phi_\rho \le c_p\rho^p$. By contrast, for $1 < p < 2$ the quadratically penalized Moreau envelope does not keep a uniformly bounded gap.

Proposition 2.2. *For $L > 0$, define the quadratically penalized Moreau envelope of ψ by*

$$\tilde{\psi}_L(r) := \inf_{z\in\mathbb{R}} \left\{ \psi(z) + \frac{L}{2}(r-z)^2 \right\}.$$

Then $\tilde{\psi}_L \le \psi$. If $p = 1$, then $\sup_r\{\psi(r) - \tilde{\psi}_L(r)\} = 1/(2L)$. If $1 < p < 2$, then $\sup_r\{\psi(r) - \tilde{\psi}_L(r)\} = \infty$.

Write $h := \psi^*$. For $1 < p < 2$, let $q := p/(p-1) > 2$, so that $h(v) = |v|^q/q$. At $p = 1$, $h = \iota_{[-1,1]}$, where ι_C is zero on C and $+\infty$ outside C. The next proposition gives the conjugate $h_\rho := \phi_\rho^*$ and its strong convexity and approximation properties.

Proposition 2.3. *For $1 \le p < 2$ and every $\rho > 0$, let $h_\rho := \phi_\rho^*$. If $1 < p < 2$, then*

$$h_\rho(v) = \begin{cases} \frac{1}{2}\rho^{2-p}v^2 & |v| \le \rho^{p-1}, \\ \frac{1}{q}|v|^q + c_p\rho^p & |v| > \rho^{p-1}. \end{cases} \tag{3}$$

At $p = 1$, $h_\rho(v) = \frac{\rho}{2}v^2 + \iota_{[-1,1]}(v)$. In both cases, h_ρ is ρ^{2-p}-strongly convex and satisfies

$$h(v) \le h_\rho(v) \le h(v) + c_p\rho^p \qquad \textit{for all } v \in \mathbb{R}. \tag{4}$$

Finally,

$$\phi_\rho(r) = \max_{v\in\mathbb{R}}\{rv - h_\rho(v)\}. \tag{5}$$

By conjugacy, the strong convexity of h_ρ gives the ρ^{p-2}-smoothness of ϕ_ρ, while the bounded additive gap gives $\psi - c_p\rho^p \le \phi_\rho \le \psi$.

For $L > 0$, define the optimal uniform approximation error over Φ_L by

$$\delta^\star(L) := \inf_{\phi\in\Phi_L} \sup_{r\in\mathbb{R}} \{\psi(r) - \phi(r)\}. \tag{6}$$

Following the analysis framework of Samakhoana & Grimmer (2026), the next theorem shows that p-Huber smoothing attains the minimum uniform approximation error over Φ_L.

Theorem 2.4. *For $1 \le p < 2$ and any $L > 0$, the p-Huber smoothing surrogate ϕ_ρ as in (2) with $\rho = L^{-\frac{1}{2-p}}$ attains the optimal uniform error $\delta^\star(L) = c_p L^{-\frac{p}{2-p}}$ among all differentiable surrogates in Φ_L.*

2.2 Composite surrogate

Applying the scalar surrogate to each residual coordinate gives the p-Huber smoothing of the loss f in (1). For a smoothing radius $\rho > 0$, define the smoothing surrogate as

$$f_\rho(\boldsymbol{u}) := \sum_{i=1}^m \phi_\rho(r_i) = \sum_{i=1}^m \phi_\rho\left((\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b})_i\right). \tag{7}$$

Moreover, consider its dual form. Let $H(\boldsymbol{v}) := \sum_{i=1}^m h(v_i)$ and $H_\rho(\boldsymbol{v}) := \sum_{i=1}^m h_\rho(v_i)$. Thus $H(\boldsymbol{v}) = \|\boldsymbol{v}\|_q^q/q$ for $1 < p < 2$, and $H = \iota_{[-1,1]^m}$ at $p = 1$. Then

$$f_\rho(\boldsymbol{u}) = \max_{\boldsymbol{v}\in\mathbb{R}^m} \{\langle \boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}, \boldsymbol{v}\rangle - H_\rho(\boldsymbol{v})\}. \tag{8}$$

One can see that $H(\boldsymbol{v}) \le H_\rho(\boldsymbol{v}) \le H(\boldsymbol{v}) + c_p m\rho^p$ for every $\boldsymbol{v} \in \mathbb{R}^m$. The maximizer is unique and equals $\boldsymbol{v}_\rho(\boldsymbol{u}) := \phi_\rho'(\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b})$ coordinatewise; hence $\nabla f_\rho(\boldsymbol{u}) = \boldsymbol{A}^\top \boldsymbol{v}_\rho(\boldsymbol{u})$.

Proposition 2.5. *The function f_ρ is convex and continuously differentiable, with $\mathrm{Lip}(\nabla f_\rho) \le \|\boldsymbol{A}\|^2\rho^{p-2}$. The approximation error is bounded as $0 \le f(\boldsymbol{u}) - f_\rho(\boldsymbol{u}) \le c_p m\rho^p$ for all $\boldsymbol{u} \in \mathbb{R}^n$. Moreover, the surrogate f_ρ is non-increasing in ρ, i.e., for any $0 < \rho_2 \le \rho_1$, any $\boldsymbol{u} \in \mathbb{R}^n$,*

$$0 \le f_{\rho_2}(\boldsymbol{u}) - f_{\rho_1}(\boldsymbol{u}) \le c_p m(\rho_1^p - \rho_2^p). \tag{9}$$

Adding ω gives the composite surrogate

$$F_\rho(\boldsymbol{u}) := f_\rho(\boldsymbol{u}) + \omega(\boldsymbol{u}). \tag{10}$$

In the dual representation (8), smoothing replaces H by H_ρ, while ω remains unchanged.

Proposition 2.6. *For every $\rho > 0$, the function F_ρ is proper, lower semicontinuous, and convex. For every $\boldsymbol{u} \in \mathrm{dom}(\omega)$, $0 \le F(\boldsymbol{u}) - F_\rho(\boldsymbol{u}) \le c_p m\rho^p$. For any $0 < \rho_2 \le \rho_1$ and $\boldsymbol{u} \in \mathrm{dom}(\omega)$, $0 \le F_{\rho_2}(\boldsymbol{u}) - F_{\rho_1}(\boldsymbol{u}) \le c_p m\big(\rho_1^p - \rho_2^p\big)$. Moreover, for every $\boldsymbol{u} \in \mathrm{dom}(\omega)$ and $\boldsymbol{u}^\star \in \mathcal{U}^\star$, $F(\boldsymbol{u}) - F^\star \le c_p m\rho^p + F_\rho(\boldsymbol{u}) - F_\rho(\boldsymbol{u}^\star)$.*

3 Optimal Deterministic Matrix–Vector Complexity

We apply accelerated proximal gradient to the p-Huber surrogate and analyze its convergence under fixed and variable smoothing schedules. Subsection 3.1 specifies the algorithm, and Subsection 3.2 analyzes the fixed-smoothing and variable-smoothing variants. Subsection 3.3 establishes a lower complexity bound. Together with the upper bound, it determines the optimal deterministic matrix–vector complexity of ℓ_p regression for $1 \le p < 2$.

3.1 Accelerated proximal gradient with p-Huber smoothing

Algorithm 1 applies accelerated proximal gradient to the p-Huber surrogate using Nesterov's three-sequence scheme (Nesterov, 2005) and a non-increasing sequence of positive smoothing radii $\{\rho_t\}_{t=0}^T$. Write $f_t := f_{\rho_t}$, $F_t := F_{\rho_t}$, and $L_t := \|\boldsymbol{A}\|^2\rho_t^{p-2}$. For the upper-bound algorithms and proofs below, a known upper bound may be used in place of $\|\boldsymbol{A}\|$, with the corresponding guarantees expressed in terms of that bound. For $t = 1, \ldots, T$, define $\Gamma_t := t(t+1)/2$ and $\beta_t := \alpha_t\gamma_t\Gamma_t$, where α_t and γ_t are the coefficients in Algorithm 1, and set $\Gamma_0 = 0$. The following lemma gives a one-step bound used in the convergence analysis of both fixed and variable smoothing.

Algorithm 1: Accelerated proximal gradient with p-Huber smoothing

Input: forward/adjoint access to $\boldsymbol{A}$, $\boldsymbol{b}$, $p \in [1, 2)$, initial point $\boldsymbol{u}_0$, positive smoothing radii $\{\rho_t\}_{t=0}^{T}$, and positive parameters $\{\gamma_t\}_{t=1}^{T}$

Initialize: $\boldsymbol{y}_0 \leftarrow \boldsymbol{z}_0 \leftarrow \boldsymbol{u}_0$

for $t = 1, \dots, T$ **do**

- $\alpha_t \leftarrow 2/(t+1)$
- $\boldsymbol{x}_t \leftarrow (1-\alpha_t)\boldsymbol{y}_{t-1} + \alpha_t \boldsymbol{z}_{t-1}$
- $\boldsymbol{g}_t \leftarrow \nabla f_{\rho_t}(\boldsymbol{x}_t) = \boldsymbol{A}^\top \phi'_{\rho_t}(\boldsymbol{A}\boldsymbol{x}_t - \boldsymbol{b})$
- $\boldsymbol{z}_t \leftarrow \mathrm{prox}_{\omega/\gamma_t}(\boldsymbol{z}_{t-1} - \gamma_t^{-1}\boldsymbol{g}_t)$
- $\boldsymbol{y}_t \leftarrow (1-\alpha_t)\boldsymbol{y}_{t-1} + \alpha_t \boldsymbol{z}_t$

Output: $\boldsymbol{y}_T$

Lemma 3.1. *Suppose that $\gamma_t \geq \alpha_t L_t$ and $\rho_t \leq \rho_{t-1}$ for all $t = 1, \dots, T$. Then, for every $\boldsymbol{u} \in \mathrm{dom}(\omega)$ and $t = 1, \dots, T$, Algorithm 1 satisfies*

$$\Gamma_t\big(F_t(\boldsymbol{y}_t) - F_t(\boldsymbol{u})\big) \;\leq\; \Gamma_{t-1}\big(F_{t-1}(\boldsymbol{y}_{t-1}) - F_{t-1}(\boldsymbol{u})\big) + \frac{\beta_t}{2}\left(\|\boldsymbol{u} - \boldsymbol{z}_{t-1}\|^2 - \|\boldsymbol{u} - \boldsymbol{z}_t\|^2\right) + c_p m \Gamma_{t-1}\big(\rho_{t-1}^p - \rho_t^p\big). \tag{11}$$

The proof combines the smooth upper model for f_t, convexity of f_t and ω, the proximal three-point inequality, and the composite drift bound in Proposition 2.6; see Appendix A.2.1.

3.2 Upper bound on matrix–vector complexity

The upper-bound analysis considers fixed and variable p-Huber smoothing. Fixed smoothing chooses its radius for a prescribed terminal horizon T. Variable smoothing uses a decreasing radius sequence independent of the terminal horizon, with a convergence guarantee under a bounded-iterate assumption.

Fixed smoothing. The fixed-smoothing variant optimizes a single surrogate. The step-size choice below makes β_t constant, so the distance terms in Lemma 3.1 telescope exactly.

Theorem 3.2. *Let $1 \leq p < 2$ and $T \geq 1$, suppose $\|\boldsymbol{A}\| > 0$, and let $D > 0$ satisfy $D \geq \mathrm{dist}(\boldsymbol{u}_0, \mathcal{U}^\star)$. Set*

$$\rho_T := \frac{2\|\boldsymbol{A}\| D}{\sqrt{mT(T+1)}}, \qquad L_T := \|\boldsymbol{A}\|^2 \rho_T^{p-2}, \tag{12}$$

and run Algorithm 1 with $\rho_t \equiv \rho_T$ and $\gamma_t := \frac{2L_T}{t}$, for $t = 1, \dots, T$. Then, the output $\boldsymbol{y}_T$ satisfies

$$F(\boldsymbol{y}_T) - F^\star \leq \frac{2^p}{p}\, m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{\sqrt{T(T+1)}}\right)^p \leq \frac{2^p}{p}\, m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{T}\right)^p. \tag{13}$$

The rate reflects the optimal scalar error–smoothness trade-off: the optimization and smoothing error bounds have orders $\|\boldsymbol{A}\|^2 D^2 \rho^{p-2}/T^2$ and $m\rho^p$, respectively. Balancing these terms yields the rate $\mathcal{O}(T^{-p})$. Prior knowledge of the terminal horizon can be removed by a restart-doubling scheme without changing the order of the rate; details are given in Appendix B.1.

Variable smoothing. The variable-smoothing variant uses the radius schedule $\rho_t := \bar{\rho}/(t+1)$, which does not require prior knowledge of the total iteration count T. Since $L_t = \|\boldsymbol{A}\|^2 \bar{\rho}^{p-2}(t+1)^{2-p}$, the upper bound on the gradient Lipschitz constant grows as $(t+1)^{2-p}$, while the uniform approximation error bound decreases as $(t+1)^{-p}$.

Theorem 3.3. *Let $1 \leq p < 2$ and $T \geq 1$, and suppose $\|\boldsymbol{A}\| > 0$. Suppose further that for some minimizer $\boldsymbol{u}^\star \in \mathcal{U}^\star$ and some $R > 0$, the accelerated sequence satisfies $\|\boldsymbol{z}_t - \boldsymbol{u}^\star\| \leq R$ for all $t = 0, \dots, T$. Set*

$$\bar{\rho} := 2\sqrt{\frac{2-p}{(p+2)m}}\,\|\boldsymbol{A}\| R, \qquad \rho_t := \frac{\bar{\rho}}{t+1}, \qquad L_t := \|\boldsymbol{A}\|^2 \rho_t^{p-2}, \tag{14}$$

and run Algorithm 1 with $\gamma_t := \alpha_t L_t$. Then, after T iterations,

$$F(\boldsymbol{y}_T) - F^\star \le C_p\, m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| R}{T+1}\right)^p, \qquad C_p := \frac{4}{p}\left(\frac{p+2}{4(2-p)}\right)^{\frac{2-p}{2}}. \tag{15}$$

For either variant, T iterations of Algorithm 1 use $2T$ individual matrix–vector products and T proximal evaluations.

Practical variants. In practice, there are several heuristic ways to further improve the performance. First, the backtracking variant replaces the global bound L_t by a local estimate $\widehat{L}_t$, accepting a candidate whenever

$$f_t(\boldsymbol{y}_t) \le f_t(\boldsymbol{x}_t) + \langle \nabla f_t(\boldsymbol{x}_t),\, \boldsymbol{y}_t - \boldsymbol{x}_t\rangle + \frac{\widehat{L}_t}{2}\|\boldsymbol{y}_t - \boldsymbol{x}_t\|^2. \tag{16}$$

Appendix B.3 gives additional conditions under which this variant retains the variable-smoothing convergence bound.

Second, for problem (1) with $\omega \equiv 0$, we also consider initialization at a least-squares solution $\boldsymbol{u}_{\mathrm{OLS}} \in \arg\min_{\boldsymbol{u}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_2^2$. This allows the method to initialize without prior knowledge of the distance D. If $\boldsymbol{A}\boldsymbol{u}_{\mathrm{OLS}} = \boldsymbol{b}$, the problem is solved. Otherwise, the variable-smoothing variant starts at $\boldsymbol{u}_{\mathrm{OLS}}$ and sets $\bar{\rho} = 2\|\boldsymbol{A}\boldsymbol{u}_{\mathrm{OLS}} - \boldsymbol{b}\|_\infty$. It uses $\rho_t = \bar{\rho}/(t+1)$ and the global bound L_t, without backtracking or reference-distance information. The full specification and its connection to the variable-smoothing bound are given in Appendix B.4.

3.3 Lower bound on matrix–vector complexity

The lower-bound analysis follows the framework of Ouyang & Xu (2021), and it applies to all adaptive deterministic methods in the following oracle model.

The algorithm accesses $\boldsymbol{A}$ through the paired matrix–vector oracle

$$\mathsf{O}_{\boldsymbol{A}}(\boldsymbol{u}, \boldsymbol{v}) := (\boldsymbol{A}\boldsymbol{u}, \boldsymbol{A}^\top \boldsymbol{v}), \qquad \boldsymbol{u} \in \mathbb{R}^n, \quad \boldsymbol{v} \in \mathbb{R}^m.$$

For a query budget $Q \in \mathbb{N}$, the algorithm is given $(p, m, n, \boldsymbol{b}, \boldsymbol{u}_0, D, \|\boldsymbol{A}\|, Q)$ before its first query. Each query and the final output may depend on these data, all preceding oracle responses, and all proximal evaluations made so far. Between queries, the algorithm may perform arbitrary deterministic computations and any number of exact evaluations of $\mathrm{prox}_{\omega/\gamma}$, for any $\gamma > 0$. Apart from the supplied norm $\|\boldsymbol{A}\|$, information about $\boldsymbol{A}$ is available only through the oracle. Matrix inversion, factorization, and linear-system solution are not available as unit-cost primitives.

To establish a worst-case lower bound for the composite problem class, it suffices to consider instances with $\omega \equiv 0$ in Theorem 3.4; the proof is given in Appendix A.2.4.

Theorem 3.4. *Fix $p \in [1, 2)$, $\Lambda, D > 0$, and an integer $Q \ge 1$. Suppose $m \ge 8Q + 18$ and $n \ge 4Q + 10$. Consider any adaptive deterministic algorithm initialized at $\boldsymbol{u}_0 \in \mathbb{R}^n$ and making at most Q calls to the paired oracle, with any number of exact proximal evaluations. There exist $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$ for problem (1) with $\omega \equiv 0$ such that $\|\boldsymbol{A}\| = \Lambda$, $\mathrm{dist}(\boldsymbol{u}_0, \mathcal{U}^\star) = D$, and $F^\star = 0$. On this instance, the algorithm's output $\boldsymbol{u}_Q$ satisfies*

$$F(\boldsymbol{u}_Q) - F^\star \ge \frac{1}{p}\left(\frac{\sqrt{3}\, c_0}{4\sqrt{2}}\right)^p m^{\frac{2-p}{2}} \left(\frac{\Lambda D}{2Q+5}\right)^p, \tag{17}$$

where $c_0 \in (0, 1)$ is a universal constant.

For an integer $T \ge 1$, under the theorem's assumptions with $Q = 2T$, this gives a worst-case objective gap of $\Omega(T^{-p})$ for any method using at most $2T$ individual products. Since Algorithm 1 with fixed smoothing attains the upper bound in Theorem 3.2 with $2T$ products, its deterministic matrix–vector complexity is optimal up to constants depending only on p in the stated oracle model.

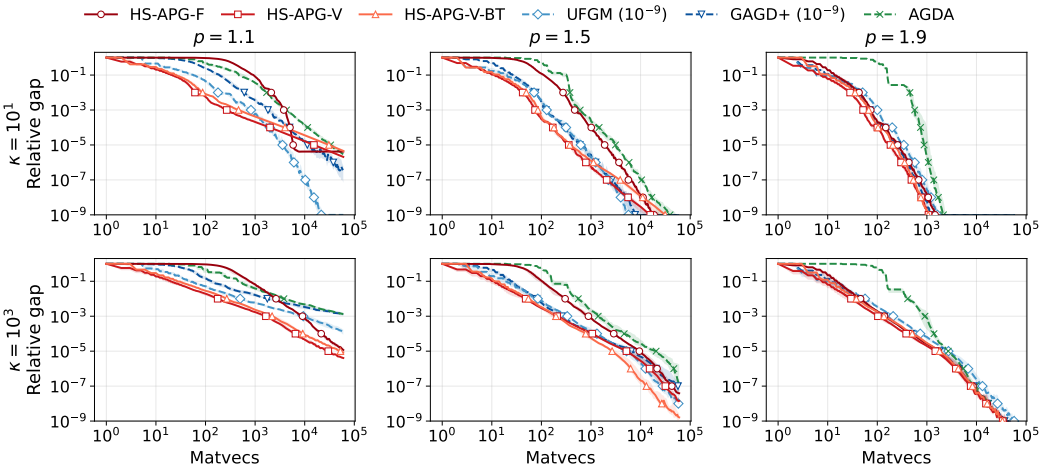

Figure 1: Convergence on synthetic instances with $p \in \{1.1, 1.5, 1.9\}$ and $\kappa(\boldsymbol{A}) \in \{10, 10^3\}$. Curves show median relative gaps over ten paired seeds against the number of matrix–vector products. Shaded bands span the 25th to 75th percentiles. UFGM and GAGD+ use target 10^{-9}. The complete 5×5 grid is shown in Figure 2 of the appendix.

4 Numerical Experiments

We compare methods based on p-Huber smoothing with first-order baselines using objective accuracy and matrix–vector product counts. Subsection 4.1 presents controlled studies on synthetic instances, Subsection 4.2 reports real-data regression experiments, and Subsection 4.3 evaluates ℓ_1-regularized sparse regression. In the first two studies, we set $\omega \equiv 0$ to examine the smoothing strategies without the additional effect of regularization.

Accuracy is measured by the relative objective gap $\mathrm{RelGap}(\boldsymbol{u}) = (F(\boldsymbol{u}) - F_{\mathrm{ref}})/(F(\boldsymbol{u}_0) - F_{\mathrm{ref}})$, where F_{ref} is a reference objective value and $\boldsymbol{u}_0$ is a common normalization point for all methods on each problem. Computational cost is measured by the number of matrix–vector products, including those used for initialization and line search. Reference construction and implementation details are provided in Appendix C.

4.1 Controlled Studies

We study how p and the conditioning of $\boldsymbol{A}$ affect the performance of fixed smoothing, variable smoothing, and variable smoothing with backtracking. We compare these methods with Nesterov's Universal Fast Gradient Method (UFGM) (Nesterov, 2015), the Accelerated Distance-Adaptive method (AGDA) (Ren et al., 2025), and Generalized-AGD+ (GAGD+) (Diakonikolas & Guzmán, 2024).

The synthetic instances cover all combinations of $p \in \{1.1, 1.25, 1.5, 1.75, 1.9\}$ and $\kappa(\boldsymbol{A}) \in \{1, 10, 10^2, 10^3, 10^4\}$. We generate ten instances for each pair, giving 250 instances in total, with $m = 200$, $n = 50$, and $\|\boldsymbol{A}\| = 1$. All methods start from $\boldsymbol{u}_0 = \boldsymbol{0}$, with initial distance $\|\boldsymbol{u}_0 - \boldsymbol{u}^\star\| = D = 10$. The construction provides the exact reference value $F_{\mathrm{ref}} = F^\star$ for each instance; details are given in Appendix C.2.1. Each method receives a maximum budget of 60,000 matrix–vector products. Fixed smoothing uses $T = 30{,}000$ to select its smoothing radius, while UFGM and GAGD+ are configured for a relative-gap target of 10^{-9}. For each method and instance, we evaluate attainment of 10^{-6} and 10^{-9} along a single run. Further algorithm settings are given in Appendix C.1. In the figures and tables, HS-APG-F, HS-APG-V, and HS-APG-V-BT denote fixed smoothing, variable smoothing, and variable smoothing with backtracking. HS-APG-OLS denotes variable smoothing with least-squares initialization and is used in the real-data experiments.

Figure 1 compares convergence for selected exponents and condition numbers; full-grid results are provided in Appendix C.2.2. Variable smoothing achieves consistently better terminal accuracy

Table 1: Real-data attainment within 60,000 algorithmic matvecs. Hits count all 18 problems. Median first-hit costs use the same subset for every method: the 15 problems at 10^{-4} and 13 at 10^{-6} where all seven methods reach the target. UFGM and GAGD+ use target-matched configurations. Costs include initialization and line search; products used only for objective reporting are excluded for every method.

	Target 10^{-4}		Target 10^{-6}	
Method	Hits / 18	Median matvecs	Hits / 18	Median matvecs
HS-APG-F	17	2,498	15	5,900
HS-APG-V	17	576	15	1,556
HS-APG-V-BT	**18**	675	15	875
HS-APG-OLS	17	**241**	**16**	**673**
UFGM	17	978	14	1,167
GAGD+	16	738	13	1,734
AGDA	16	1,564	13	2,241

than the baselines in the small-p, poorly conditioned regime, yielding lower terminal gaps on all 60 instances with $p \leq 1.25$ and $\kappa(\boldsymbol{A}) \geq 10^2$. Across the full grid, it also reaches 10^{-6} on 201 of 250 instances, the highest attainment count among the six methods. UFGM achieves lower median terminal gaps at $p = 1.1$ and $\kappa(\boldsymbol{A}) \in \{1, 10\}$, and leads at the stricter target 10^{-9}, reaching 128 of 250 instances. At $\kappa(\boldsymbol{A}) = 10^4$, high accuracy remains difficult for all methods: each reaches 10^{-9} on at most two of the 50 instances (Table 3).

The variable-smoothing methods generally achieve lower terminal gaps than fixed smoothing, but neither variant uniformly dominates the other. Without backtracking, variable smoothing has lower terminal gaps on all $p = 1.1$ instances, whereas backtracking lowers the median terminal gap at $p = 1.5$ and $\kappa(\boldsymbol{A}) = 10^3$. Backtracking can permit larger step sizes, but its line search requires additional matrix–vector products within the same budget.

4.2 Real-world data evaluations

We evaluate the methods on six UCI regression datasets (Power, Year, Protein, KEGG, CT-Slices, and Buzz) with $p \in \{1.1, 1.5, 1.9\}$, yielding 18 optimization problems. Since D may be unknown in practice, we also test a least-squares-initialized variant of variable smoothing. It sets the smoothing scale from the least-squares residual and requires no reference-distance information. The least-squares point is computed from zero, while all other methods start at zero. The other smoothing variants use a reference-distance proxy.

Each method receives a budget of 60,000 matrix–vector products, including initialization and line search. Fixed smoothing uses $T = 30{,}000$ to set its radius. We compare attainment at relative-gap targets 10^{-4} and 10^{-6}. UFGM and GAGD+ are configured for the accuracy target used in each comparison. Dataset sources, preprocessing, reference certification, and implementation details are given in Appendix C.3.

Table 1 shows that the smoothing methods reach each target on at least as many problems as the baselines, while variable smoothing has lower median costs than fixed smoothing. Backtracking reaches 10^{-4} on all 18 problems. At 10^{-6}, each smoothing variant reaches at least 15 problems, compared with 13 or 14 for the baselines. Variable smoothing with and without backtracking has lower median costs than all three baselines at 10^{-4}; with backtracking, this advantage also holds at 10^{-6}. These cost comparisons use the common subsets where all seven methods reach the corresponding target. The least-squares-initialized variant reaches 17 and 16 problems at the two targets, respectively, with low median costs including initialization. It provides a practical option when D is unknown. Tables 6 and 7 provide problem-level results, while Appendix C.3.4 presents convergence curves and a separate gradient-restart ablation.

4.3 ℓ_1-REGULARIZED SPARSE REGRESSION

We evaluate the composite formulation on ℓ_1-regularized regression using E2006 financial-report text (Kogan et al., 2009). The smoothing methods apply the p-Huber surrogate to the loss and handle the penalty by soft thresholding. We use the TFIDF and log1p representations with $p = 1.5$, an unpenalized intercept, and $\lambda/\lambda_{\max} \in \{0.01, 0.03, 0.1, 0.3\}$, giving eight optimization problems. Here, $\lambda_{\max}$ is the smallest penalty for which setting all feature coefficients to zero is optimal (Hastie et al., 2015); its precise definition is given in (92) in Appendix C.4.1. We omit HS-APG-F in this experiment because the preceding studies already evaluate the fixed-smoothing method extensively and show that the variable-smoothing variants are generally more competitive in first-hit cost. The E2006 study therefore focuses on the practical variants HS-APG-V and HS-APG-V-BT, together with composite UFGM and GAGD+, each run separately for relative-gap targets 10^{-4} and 10^{-6}.

All methods start with zero feature coefficients and an intercept chosen to minimize the loss. Each run has a budget of $30{,}000$ matrix–vector products for TFIDF or $12{,}000$ for log1p, including line-search trials. The smoothing variants use a reference-candidate distance as the radius bound, and relative gaps are computed using independently certified reference lower bounds. Appendix C.4.1 gives the preprocessing, reference construction, and distance proxy.

Table 2 reports the terminal relative gap of every run. Variable smoothing attains the smallest terminal gap on TFIDF at ratios 0.01, 0.03, and 0.1, and on log1p at 0.01 and 0.03. Variable smoothing with backtracking attains the smallest terminal gap in the remaining cases: 0.3 on TFIDF, and 0.1 and 0.3 on log1p. The benefit of backtracking over variable smoothing is therefore concentrated at the larger penalties in this grid. No run reaches relative gap 10^{-4} within its budget. Appendix C.4.1 reports thresholded active-feature counts. We also consider box-constrained image deblurring with $\omega = \iota_{[0,1]^n}$, constraining each pixel intensity to $[0, 1]$. Appendix C.4.2 presents its setup and results.

Table 2: Terminal relative gaps for all 48 runs. Columns give $\lambda/\lambda_{\max}$. Budgets are $30{,}000$ matrix–vector products for TFIDF and $12{,}000$ for log1p. Bold entries indicate the smallest gap among the six configurations in each column within each representation.

	TFIDF			
Method / target	0.01	0.03	0.1	0.3
HS-APG-V	$\mathbf{3.0992 \times 10^{-3}}$	$\mathbf{2.0121 \times 10^{-3}}$	$\mathbf{7.8028 \times 10^{-4}}$	1.6917×10^{-3}
HS-APG-V-BT	4.9278×10^{-3}	2.9283×10^{-3}	1.7572×10^{-3}	$\mathbf{4.6715 \times 10^{-4}}$
UFGM (10^{-4})	1.2673×10^{-2}	8.7801×10^{-3}	5.9847×10^{-3}	2.8517×10^{-3}
UFGM (10^{-6})	1.2660×10^{-2}	8.7173×10^{-3}	6.0044×10^{-3}	2.8439×10^{-3}
GAGD+ (10^{-4})	6.7852×10^{-3}	3.7290×10^{-3}	9.4428×10^{-4}	7.1823×10^{-4}
GAGD+ (10^{-6})	6.7852×10^{-3}	3.7290×10^{-3}	9.4428×10^{-4}	7.1823×10^{-4}
	log1p			
Method / target	0.01	0.03	0.1	0.3
HS-APG-V	$\mathbf{1.1392 \times 10^{-2}}$	$\mathbf{3.2231 \times 10^{-2}}$	3.3366×10^{-2}	8.0258×10^{-3}
HS-APG-V-BT	1.3119×10^{-2}	3.5383×10^{-2}	$\mathbf{1.2197 \times 10^{-2}}$	$\mathbf{1.3372 \times 10^{-3}}$
UFGM (10^{-4})	2.1655×10^{-2}	5.9475×10^{-2}	8.4516×10^{-2}	5.6877×10^{-3}
UFGM (10^{-6})	2.1670×10^{-2}	5.9423×10^{-2}	8.4810×10^{-2}	5.6795×10^{-3}
GAGD+ (10^{-4})	1.6539×10^{-2}	4.4306×10^{-2}	1.6620×10^{-2}	1.9741×10^{-3}
GAGD+ (10^{-6})	1.6547×10^{-2}	4.4306×10^{-2}	1.6620×10^{-2}	1.9741×10^{-3}

5 CONCLUSION

We establish the optimal one-sided approximation–smoothness trade-off for p-Huber smoothing and apply the resulting surrogate in APG for composite ℓ_p regression. The fixed-smoothing variant attains an ε-optimal solution for $1 \le p < 2$ in $\mathcal{O}(\varepsilon^{-1/p})$ matrix–vector products, matching the deterministic lower bound in the stated oracle model. Experiments on synthetic and real data show competitive performance and illustrate the method's applicability to ℓ_1 regularization and box constraints.

AI USE STATEMENT

Generative AI tools, including OpenAI Codex, were used to assist with manuscript review and revision. Their use included reviewing the presentation and internal consistency of mathematical statements and proofs; providing feedback on the reporting and interpretation of experiments; summarizing manuscript-related research notes and meeting records; and editing prose for clarity and concision. Generative AI also assisted with assembling and anonymizing the supplementary reproducibility artifact; creating, editing, and reviewing code, tests, and plotting scripts; and regenerating figures and tables from audited frozen outputs. These uses did not replace the authors' responsibility for developing the mathematical results, conducting the experiments, or interpreting the evidence. AI-assisted suggestions and artifacts were checked against the manuscript, derivations, implementation records, experimental artifacts, and primary sources before incorporation. The authors reviewed the incorporated changes and take responsibility for the final content of the paper and supplementary materials.

REPRODUCIBILITY STATEMENT

Proofs of all stated results are collected in Appendix A. The oracle model, algorithms, and complexity bounds are specified in Sections 2 and 3. Appendix C gives the experimental settings, data processing, and evaluation protocol. All reported runs used Julia 1.12.7 on a single server with a 12-core Intel Core i7-12700 CPU and 32 GB of RAM. An anonymized code archive, submitted as supplementary material, contains the method implementations and the drivers for the controlled, real-data, E2006, and box-constrained experiments. Processed datasets, certified references, and the formal run archives are not included in that archive.

REFERENCES

Deeksha Adil and Sushant Sachdeva. Faster p-norm minimizing flows, via smoothed q-norm problems. In Shuchi Chawla (ed.), *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pp. 892–910. SIAM, 2020. doi: 10.1137/1.9781611975994.54. URL https://doi.org/10.1137/1.9781611975994.54.

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*, SODA '19, pp. 1405–1424, 2019a.

Deeksha Adil, Richard Peng, and Sushant Sachdeva. Fast, provably convergent irls algorithm for p-norm linear regression. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019b. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/46c7cb50b373877fb2f8d5c4517bb969-Paper.pdf.

Deeksha Adil, Rasmus Kyng, Richard Peng, and Sushant Sachdeva. Fast algorithms for ℓ_p-regression. *Journal of the ACM*, 71(5):1–45, 2024.

Heinz H Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2017.

Amir Beck and Marc Teboulle. Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization*, 22(2):557–580, 2012.

T. Bertin-Mahieux. Year Prediction MSD. UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C50K61.

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

Jelena Diakonikolas and Cristóbal Guzmán. Complementary composite minimization, small gradients in general norms, and applications. *Mathematical Programming*, 208(1):319–363, 2024.

Celestine Dünner, Simone Forte, Martin Takac, and Martin Jaggi. Primal-dual rates and certificates. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 783–792, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/dunner16.html.

Ahmed El Alaoui, Xiang Cheng, Aaditya Ramdas, Martin J. Wainwright, and Michael I. Jordan. Asymptotic behavior of ℓ_p-based Laplacian regularization in semi-supervised learning. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir (eds.), *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pp. 879–906, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR. URL https://proceedings.mlr.press/v49/elalaoui16.html.

Alina Ene, Ta Duy Nguyen, and Adrian Vladu. Improved ℓ_p regression via iteratively reweighted least squares. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 138568–138594, 2026.

Simon Foucart, Alain Pajor, Holger Rauhut, and Tino Ullrich. The gelfand widths of ℓ_p-balls for $0 < p \leq 1$. *Journal of Complexity*, 26(6):629–640, 2010.

F. Graf, H.-P. Kriegel, M. Schubert, S. Poelsterl, and A. Cavallaro. Relative location of CT slices on axial axis. UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C5CP6G.

William W. Hager and Hongchao Zhang. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization*, 16(1):170–192, 2005. doi: 10.1137/030601880. URL https://doi.org/10.1137/030601880.

Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

Georges Hebrail and Alice Berard. Individual Household Electric Power Consumption. UCI Machine Learning Repository, 2006. DOI: https://doi.org/10.24432/C58K54.

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*, pp. 529–542, 2022.

Arun Jambulapati, James R Lee, Yang P. Liu, and Aaron Sidford. Sparsifying generalized linear models. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pp. 1665–1675, 2024.

Franois Kawala, Ahlame Douzal, Eric Gaussier, and Eustache Diemert. Buzz in social media . UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C56G6V.

Shimon Kogan, Dimitry Levin, Bryan R. Routledge, Jacob S. Sagi, and Noah A. Smith. Predicting risk from financial reports with regression. In Mari Ostendorf, Michael Collins, Shri Narayanan, Douglas W. Oard, and Lucy Vanderwende (eds.), *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 272–280, Boulder, Colorado, June 2009. Association for Computational Linguistics. URL https://aclanthology.org/N09-1031/.

Yuying Li. A globally convergent method for ℓ_p problems. *SIAM Journal on Optimization*, 3(3): 609–629, 1993.

Meng Liu and David F Gleich. Strongly local p-norm-cut algorithms for semi-supervised learning and local graph clustering. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5023–5035. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3501672ebc68a5524629080e3ef60aef-Paper.pdf.

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.

Chris J Maddison, Daniel Paulin, Yee Whye Teh, and Arnaud Doucet. Dual space preconditioning for gradient descent. *SIAM Journal on Optimization*, 31(1):991–1016, 2021.

A. H. Money, J. F. Affleck-Graves, M. L. Hart, and G. D. I. Barr. The linear regression model: Lp norm estimation and the choice of p. *Communications in Statistics - Simulation and Computation*, 11(1):89–109, 1982.

Muhammad Naeem and Sohail Asghar. KEGG Metabolic Reaction Network (Undirected). UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C5G609.

Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1): 127–152, 2005.

Yu Nesterov. Universal gradient methods for convex optimization problems. *Mathematical Programming*, 152(1):381–404, 2015.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer New York, NY, 2006. URL `https://books.google.com.ph/books?id=VbHYoSyelFcC`.

Brendan O'Donoghue and Emmanuel Candès. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 15(3):715–732, 2015. doi: 10.1007/s10208-013-9150-3.

Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, 185(1):1–35, 2021. doi: 10.1007/s10107-019-01420-0.

Prashant Rana. Physicochemical Properties of Protein Tertiary Structure. UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C5QW3H.

Yijin Ren, Haifeng Xu, and Qi Deng. Accelerated distance-adaptive methods for hölder smooth and convex optimization. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 165266–165311. Curran Associates, Inc., 2025. doi: 10.52202/085713-5510. URL `https://proceedings.neurips.cc/paper_files/paper/2025/file/f17e298485951aeda6b94125c1f66460-Paper-Conference.pdf`.

Thabo Samakhoana and Benjamin Grimmer. The optimal smoothings of sublinear functions and convex cones. *Set-Valued and Variational Analysis*, 34(2):24, 2026.

Stéfan Van der Walt, Johannes L Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: image processing in python. *PeerJ*, 2:e453, 2014.

A Omitted Proofs from the Main Body

In this appendix, we collect the proofs of all results stated in the main body.

A.1 Proofs for Section 2

A.1.1 An auxiliary coordinatewise lifting lemma

Lemma A.1 (Coordinatewise lifting)**.** *Let $L > 0$ and let $\phi \in \mathcal{C}^1(\mathbb{R})$ be convex,* $\mathrm{Lip}(\phi') \le L$*, and suppose $0 \le \psi(r) - \phi(r) \le \delta$ for all $r \in \mathbb{R}$. Then*

$$f_\phi(\boldsymbol{u}) := \sum_{i=1}^{m} \phi\big((\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b})_i\big)$$

is convex and continuously differentiable, with $\mathrm{Lip}(\nabla f_\phi) \le L\|\boldsymbol{A}\|^2$ *and $0 \le f(\boldsymbol{u}) - f_\phi(\boldsymbol{u}) \le m\delta$ for all $\boldsymbol{u} \in \mathbb{R}^n$.*

Proof. Convexity follows because each summand is the composition of a convex scalar function with an affine map. The chain rule gives

$$\nabla f_\phi(\boldsymbol{u}) = \boldsymbol{A}^\top \phi'(\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}),$$

where ϕ' is applied coordinatewise. Thus, for all $\boldsymbol{u}, \boldsymbol{w} \in \mathbb{R}^n$,

$$\begin{aligned}\|\nabla f_\phi(\boldsymbol{u}) - \nabla f_\phi(\boldsymbol{w})\| &\le \|\boldsymbol{A}\|\,\|\phi'(\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}) - \phi'(\boldsymbol{A}\boldsymbol{w} - \boldsymbol{b})\| \\ &\le L\|\boldsymbol{A}\|\,\|\boldsymbol{A}(\boldsymbol{u} - \boldsymbol{w})\| \le L\|\boldsymbol{A}\|^2\|\boldsymbol{u} - \boldsymbol{w}\|.\end{aligned}$$

Finally, summing $0 \le \psi(r_i) - \phi(r_i) \le \delta$ over the m residual coordinates yields the uniform approximation bound. □

A.1.2 Proof of Proposition 2.2

Proof. The envelope $\tilde{\psi}_L$ is convex and has an L-Lipschitz derivative, and $\tilde{\psi}_L \le \psi$ because $z = r$ is feasible. For $r > 0$ it is enough to minimize over $z \ge 0$; the case $r < 0$ follows by evenness.

If $p = 1$, the minimizer is $z = 0$ when $0 \le r \le 1/L$, and $z = r - 1/L$ when $r > 1/L$. In the first range the gap is $r - (L/2)r^2 \le 1/(2L)$. In the second range it equals

$$r - \big(r - 1/L\big) - (L/2)(1/L)^2 = 1/(2L).$$

Thus the supremum is $1/(2L)$.

Now let $1 < p < 2$. For each $z > 0$ set $r = z + z^{p-1}/L$. This z satisfies the first-order condition for $\tilde{\psi}_L(r)$, so

$$\psi(r) - \tilde{\psi}_L(r) = \frac{r^p - z^p}{p} - \frac{z^{2p-2}}{2L}.$$

The mean-value theorem gives $r^p - z^p = p\xi^{p-1}(r - z)$ for some $\xi \in (z, r)$. Since $r - z = z^{p-1}/L$ and $\xi > z$,

$$\psi(r) - \tilde{\psi}_L(r) > z^{p-1} \cdot \frac{z^{p-1}}{L} - \frac{z^{2p-2}}{2L} = \frac{z^{2p-2}}{2L}.$$

The right-hand side tends to infinity as $z \to \infty$, and the corresponding r does as well. Therefore the supremum is infinite. □

A.1.3 Proof of Proposition 2.3

Proof. At $p = 1$, the conjugate of $\psi(r) = |r|$ is $h = \iota_{[-1,1]}$. If $|v| > 1$, the expression $vr - \phi_\rho(r)$ tends to $+\infty$ as r tends to infinity with the sign of v. If $|v| \le 1$, the supremum is attained at $r = \rho v$ and equals $\rho v^2/2$. This proves the stated endpoint formula. Subtracting $\rho v^2/2$ from h_ρ leaves the convex indicator $\iota_{[-1,1]}$, so h_ρ is ρ-strongly convex. On $[-1, 1]$, its correction to h lies in $[0, \rho/2]$; outside this interval, both functions are $+\infty$. Thus (4) holds as an extended-real inequality.

Now let $1 < p < 2$. Both ϕ_ρ and its conjugate are even, so it suffices to take $v \geq 0$. By definition,

$$h_\rho(v) = \sup_{r \geq 0}\{vr - \phi_\rho(r)\}.$$

If the maximizer lies in $0 \leq r \leq \rho$, the first-order condition is $v = \rho^{p-2}r$, hence $r = \rho^{2-p}v$. This point belongs to the inner branch exactly when $v \leq \rho^{p-1}$, and substitution gives $h_\rho(v) = \frac{1}{2}\rho^{2-p}v^2$. If the maximizer lies in $r > \rho$, the first-order condition is $v = r^{p-1}$, hence $r = v^{q-1}$. This point belongs to the outer branch exactly when $v > \rho^{p-1}$, and

$$h_\rho(v) = v^q - \frac{1}{p}v^q + c_p\rho^p = \frac{1}{q}v^q + c_p\rho^p.$$

At $v = \rho^{p-1}$, the two expressions agree because $1/q + c_p = 1/2$. This proves (3).

On the inner branch, $h''_\rho(v) = \rho^{2-p}$. On the outer branch,

$$h''_\rho(v) = (q-1)|v|^{q-2} \geq (q-1)\rho^{(p-1)(q-2)} = (q-1)\rho^{2-p} \geq \rho^{2-p}.$$

Since the first derivatives match at the two junctions, h_ρ is ρ^{2-p}-strongly convex on $\mathbb{R}$.

It remains to prove the uniform correction bound. For $|v| \geq \rho^{p-1}$, the difference $h_\rho(v) - h(v)$ equals $c_p\rho^p$. For $|v| \leq \rho^{p-1}$, write $s := |v|/\rho^{p-1} \in [0,1]$. Then

$$h_\rho(v) - h(v) = \rho^p\left(\frac{1}{2}s^2 - \frac{1}{q}s^q\right).$$

The function in parentheses has derivative $s - s^{q-1} \geq 0$ on $[0,1]$, and therefore increases from zero to $1/2 - 1/q = c_p$. This gives (4). For either case, ϕ_ρ is finite, closed, and convex, so $\phi_\rho = h^*_\rho$ by biconjugacy, proving (5). □

A.1.4 Proof of Proposition 2.5

Proof. By evenness it suffices to consider $r \geq 0$. At $r = \rho$, the two branches in (2) have the common value $\rho^p/2$ and the common derivative ρ^{p-1}. Hence $\phi_\rho \in \mathcal{C}^1(\mathbb{R})$, with derivative $\phi'_\rho(r) = (\max\{\rho, |r|\})^{p-2}r$. On $(0,\rho)$, its second derivative is ρ^{p-2}; on (ρ,∞), it is $(p-1)r^{p-2}$, which belongs to $[0,(p-1)\rho^{p-2}]$. Therefore the derivative is nondecreasing and ρ^{p-2}-Lipschitz. The Lipschitz constant is exact because the inner branch has slope ρ^{p-2}.

Let $e_\rho(r) := \psi(r) - \phi_\rho(r)$. For $0 < r \leq \rho$,

$$e'_\rho(r) = r^{p-1} - \rho^{p-2}r = r\big(r^{p-2} - \rho^{p-2}\big) \geq 0,$$

because $p - 2 < 0$, while $e_\rho(0) = 0$. Thus e_ρ increases from zero to $e_\rho(\rho) = c_p\rho^p$; for $r > \rho$, it is identically $c_p\rho^p$. Hence $0 \leq \psi(r) - \phi_\rho(r) \leq c_p\rho^p$ for every $r \in \mathbb{R}$, as claimed.

For the radius drift, fix r and differentiate with respect to ρ. Away from $\rho = |r|$,

$$\partial_\rho\phi_\rho(r) = \begin{cases} -\dfrac{2-p}{2}\rho^{p-1}, & 0 < \rho < |r|, \\ -\dfrac{2-p}{2}\rho^{p-3}r^2, & \rho > |r|. \end{cases}$$

The two expressions agree at $\rho = |r|$, and in either case

$$-\frac{2-p}{2}\rho^{p-1} \leq \partial_\rho\phi_\rho(r) \leq 0.$$

Therefore, for $0 < \rho_2 \leq \rho_1$, integration from ρ_2 to ρ_1 gives

$$0 \leq \phi_{\rho_2}(r) - \phi_{\rho_1}(r) \leq c_p\big(\rho_1^p - \rho_2^p\big).$$

Applying Lemma A.1 with $L = \rho^{p-2}$ and $\delta = c_p\rho^p$ establishes the stated convexity, smoothness, and approximation properties of f_ρ. Finally, summing the last scalar inequality over the m residual coordinates gives the claimed radius-drift bound for f_ρ. □

A.1.5 Proof of Proposition 2.6

Proof. By Proposition 2.5, f_ρ is finite, continuous, and convex. Since ω is proper, lower semicontinuous, and convex, adding the finite continuous function f_ρ shows that F_ρ is proper, lower semicontinuous, and convex.

For every $\boldsymbol{u} \in \mathrm{dom}(\omega)$, the regularizer is finite and cancels from the relevant differences. Therefore

$$F(\boldsymbol{u}) - F_\rho(\boldsymbol{u}) = f(\boldsymbol{u}) - f_\rho(\boldsymbol{u}),$$

so the claimed approximation bound follows from Proposition 2.5. Likewise, for $0 < \rho_2 \le \rho_1$,

$$F_{\rho_2}(\boldsymbol{u}) - F_{\rho_1}(\boldsymbol{u}) = f_{\rho_2}(\boldsymbol{u}) - f_{\rho_1}(\boldsymbol{u}),$$

and the radius-drift bound follows from the same proposition.

Finally, let $\boldsymbol{u}^\star \in \mathcal{U}^\star$. Since $F(\boldsymbol{u}^\star) = F^\star$, adding and subtracting $f_\rho(\boldsymbol{u})$ and $f_\rho(\boldsymbol{u}^\star)$ gives

$$\begin{aligned} F(\boldsymbol{u}) - F^\star = f(\boldsymbol{u}) - f_\rho(\boldsymbol{u}) + F_\rho(\boldsymbol{u}) - F_\rho(\boldsymbol{u}^\star) \\ + f_\rho(\boldsymbol{u}^\star) - f(\boldsymbol{u}^\star). \end{aligned}$$

The claimed inequality follows from

$$f(\boldsymbol{u}) - f_\rho(\boldsymbol{u}) \le c_p m \rho^p \quad \text{and} \quad f_\rho(\boldsymbol{u}^\star) - f(\boldsymbol{u}^\star) \le 0.$$

□

A.1.6 Proof of Theorem 2.4

Proof. Fix $L > 0$ and $\phi \in \Phi_L$, and let $\delta := \sup_r(\psi(r) - \phi(r))$. The conclusion is immediate if $\delta = +\infty$, so suppose $\delta < \infty$. The standard upper model for a function with L-Lipschitz derivative gives, for every $r > 0$,

$$\phi(r) \le \phi(0) + \phi'(0)r + \frac{L}{2}r^2,$$

$$\phi(-r) \le \phi(0) - \phi'(0)r + \frac{L}{2}r^2.$$

Since $\phi(0) \le \psi(0) = 0$, averaging these inequalities and using the definition of δ yields

$$\delta \ge \frac{\psi(r) - \phi(r) + \psi(-r) - \phi(-r)}{2} \ge \frac{r^p}{p} - \frac{Lr^2}{2}.$$

The right-hand side is maximized at $r = L^{-1/(2-p)}$, where its value is $c_p L^{-p/(2-p)}$. Hence $\delta^\star(L) \ge c_p L^{-p/(2-p)}$.

Conversely, choose $\rho = L^{-1/(2-p)}$. Then $\rho^{p-2} = L$. The scalar properties established above show that ϕ_ρ lies below ψ, has L-Lipschitz derivative, and has worst-case error $c_p\rho^p = c_p L^{-p/(2-p)}$. Thus $\phi_\rho \in \Phi_L$, proving equality and attainment.

For completeness, the same constant has a direct dual derivation on the convex subclass. If $\phi \in \Phi_L$ is convex, set $\mu := L^{-1}$, $g := \phi^*$, and $\delta := \sup_r(\psi(r) - \phi(r))$. Then g is μ-strongly convex and $h \le g \le h + \delta$. At $p = 1$, these inequalities imply $\mathrm{dom}\, g = [-1, 1]$ and

$$\delta \ge \frac{g(1) + g(-1)}{2} \ge g(0) + \frac{\mu}{2} \ge \frac{\mu}{2} = \frac{1}{2L}.$$

For $1 < p < 2$, strong convexity gives, for every $a > 0$,

$$\frac{a^q}{q} + \delta \ge \frac{g(a) + g(-a)}{2} \ge g(0) + \frac{\mu a^2}{2} \ge \frac{\mu a^2}{2}.$$

Therefore

$$\delta \ge \frac{\mu a^2}{2} - \frac{a^q}{q}.$$

Maximizing the right-hand side at $a = \mu^{1/(q-2)}$ gives

$$\delta \ge \left(\frac{1}{2} - \frac{1}{q}\right)\mu^{q/(q-2)} = c_p L^{-p/(2-p)}.$$

For $\mu = \rho^{2-p}$, the maximizing dual point is $a = \rho^{p-1}$, exactly the breakpoint in (3). □

A.2 Proofs for Section 3

A.2.1 Proof of Lemma 3.1

Proof. The proximal subproblem is strongly convex, so its minimizer is unique and belongs to $\mathrm{dom}(\omega)$. Since $\mathrm{dom}(\omega)$ is convex and $\boldsymbol{y}_0 = \boldsymbol{z}_0 = \boldsymbol{u}_0 \in \mathrm{dom}(\omega)$, all iterates $\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{z}_t$ lie in $\mathrm{dom}(\omega)$.

Fix t and $\boldsymbol{u} \in \mathrm{dom}(\omega)$, and write $\boldsymbol{g}_t := \nabla f_t(\boldsymbol{x}_t)$. Smoothness of f_t and $\boldsymbol{y}_t - \boldsymbol{x}_t = \alpha_t(\boldsymbol{z}_t - \boldsymbol{z}_{t-1})$ give

$$f_t(\boldsymbol{y}_t) \le f_t(\boldsymbol{x}_t) + \alpha_t \langle \boldsymbol{g}_t, \boldsymbol{z}_t - \boldsymbol{z}_{t-1} \rangle + \frac{L_t \alpha_t^2}{2} \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|^2. \tag{18}$$

Convexity of f_t, applied at $\boldsymbol{x}_t$ to $\boldsymbol{y}_{t-1}$ and $\boldsymbol{u}$, yields

$$f_t(\boldsymbol{x}_t) + \alpha_t \langle \boldsymbol{g}_t, \boldsymbol{u} - \boldsymbol{z}_{t-1} \rangle \le (1-\alpha_t) f_t(\boldsymbol{y}_{t-1}) + \alpha_t f_t(\boldsymbol{u}). \tag{19}$$

Convexity of ω also gives

$$\omega(\boldsymbol{y}_t) \le (1-\alpha_t)\omega(\boldsymbol{y}_{t-1}) + \alpha_t \omega(\boldsymbol{z}_t). \tag{20}$$

Proximal optimality provides $\boldsymbol{s}_t \in \partial\omega(\boldsymbol{z}_t)$ such that

$$\boldsymbol{g}_t + \boldsymbol{s}_t + \gamma_t(\boldsymbol{z}_t - \boldsymbol{z}_{t-1}) = \boldsymbol{0}.$$

The subgradient inequality and the polarization identity therefore imply

$$\begin{aligned} &\langle \boldsymbol{g}_t, \boldsymbol{z}_t - \boldsymbol{u} \rangle + \omega(\boldsymbol{z}_t) - \omega(\boldsymbol{u}) \\ &\quad \le \frac{\gamma_t}{2} \left(\|\boldsymbol{u} - \boldsymbol{z}_{t-1}\|^2 - \|\boldsymbol{u} - \boldsymbol{z}_t\|^2 - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|^2 \right). \end{aligned} \tag{21}$$

Combining (18)– (21) gives

$$\begin{aligned} F_t(\boldsymbol{y}_t) - F_t(\boldsymbol{u}) \le (1-\alpha_t)\big(F_t(\boldsymbol{y}_{t-1}) - F_t(\boldsymbol{u})\big) \\ + \frac{\alpha_t \gamma_t}{2} \left(\|\boldsymbol{u} - \boldsymbol{z}_{t-1}\|^2 - \|\boldsymbol{u} - \boldsymbol{z}_t\|^2 \right) \\ - \frac{\alpha_t}{2}(\gamma_t - \alpha_t L_t)\|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|^2. \end{aligned} \tag{22}$$

The last term is nonpositive. Multiplying by Γ_t and using $\Gamma_{t-1} = (1-\alpha_t)\Gamma_t$ and $\beta_t = \alpha_t \gamma_t \Gamma_t$ gives the same-surrogate estimate

$$\begin{aligned} \Gamma_t\big(F_t(\boldsymbol{y}_t) - F_t(\boldsymbol{u})\big) \le \Gamma_{t-1}\big(F_t(\boldsymbol{y}_{t-1}) - F_t(\boldsymbol{u})\big) \\ + \frac{\beta_t}{2} \left(\|\boldsymbol{u} - \boldsymbol{z}_{t-1}\|^2 - \|\boldsymbol{u} - \boldsymbol{z}_t\|^2 \right). \end{aligned} \tag{23}$$

Finally, $\rho_t \le \rho_{t-1}$ and the composite radius-drift bound in Proposition 2.6 give

$$F_t(\boldsymbol{y}_{t-1}) - F_t(\boldsymbol{u}) \le F_{t-1}(\boldsymbol{y}_{t-1}) - F_{t-1}(\boldsymbol{u}) + c_p m(\rho_{t-1}^p - \rho_t^p).$$

Substituting this into (23) proves (11). □

A.2.2 Proof of Theorem 3.2

Proof. Choose $\boldsymbol{u}^\star \in \mathcal{U}^\star$ with $\|\boldsymbol{u}_0 - \boldsymbol{u}^\star\| \le D$. For the fixed radius, the drift term in Lemma 3.1 vanishes. Moreover,

$$\gamma_t = \frac{2L_T}{t} \ge \frac{2L_T}{t+1} = \alpha_t L_T, \qquad \beta_t = \frac{2}{t+1}\frac{2L_T}{t}\frac{t(t+1)}{2} = 2L_T.$$

Summing (11) from $t = 1$ to T, using $\Gamma_0 = 0$, $\boldsymbol{z}_0 = \boldsymbol{u}_0$, and the constancy of β_t, gives

$$\Gamma_T\big(F_{\rho_T}(\boldsymbol{y}_T) - F_{\rho_T}(\boldsymbol{u}^\star)\big) \le L_T\big(\|\boldsymbol{u}^\star - \boldsymbol{u}_0\|^2 - \|\boldsymbol{u}^\star - \boldsymbol{z}_T\|^2\big) \le L_T D^2.$$

Combining this estimate with the objective-gap bound in Proposition 2.6 and $\Gamma_T = T(T+1)/2$ yields, for every fixed $\rho > 0$,

$$F(\boldsymbol{y}_T) - F^\star \le c_p m \rho^p + \frac{2\|\boldsymbol{A}\|^2 D^2}{T(T+1)} \rho^{p-2}. \tag{24}$$

The derivative of the right-hand side vanishes uniquely at

$$\rho^2 = \frac{4\|\boldsymbol{A}\|^2 D^2}{mT(T+1)},$$

which is (12). At this radius, the two first-order balance relations give

$$F(\boldsymbol{y}_T) - F^\star \le \frac{m}{p}\rho_T^p = \frac{2^p}{p} m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{\sqrt{T(T+1)}}\right)^p.$$

The final inequality in (13) follows from $T(T+1) \ge T^2$. Each iteration evaluates $\boldsymbol{A}\boldsymbol{x}_t$, one adjoint product, and one proximal map, establishing the stated counts. □

A.2.3 Proof of Theorem 3.3

Proof. Let $a_t := \|\boldsymbol{z}_t - \boldsymbol{u}^\star\|^2$. Under $\rho_t = \bar{\rho}/(t+1)$ and $\gamma_t = \alpha_t L_t$,

$$\beta_t = \alpha_t^2 L_t \Gamma_t = \frac{2t}{t+1}\|\boldsymbol{A}\|^2 \bar{\rho}^{p-2}(t+1)^{2-p} = 2\|\boldsymbol{A}\|^2 \bar{\rho}^{p-2} t(t+1)^{1-p}. \tag{25}$$

The function $s \mapsto s(s+1)^{1-p}$ is increasing on $(0,\infty)$, because its derivative is $(s+1)^{-p}[1 + (2-p)s] > 0$. Hence $\{\beta_t\}$ is non-decreasing.

Summing Lemma 3.1 with $\boldsymbol{u} = \boldsymbol{u}^\star$ gives

$$\Gamma_T\big(F_T(\boldsymbol{y}_T) - F_T(\boldsymbol{u}^\star)\big) \le \frac{1}{2}\sum_{t=1}^{T} \beta_t(a_{t-1} - a_t) + c_p m \sum_{t=1}^{T} \Gamma_{t-1}(\rho_{t-1}^p - \rho_t^p). \tag{26}$$

By Abel summation and $a_t \le R^2$,

$$\sum_{t=1}^{T} \beta_t(a_{t-1} - a_t) = \beta_1 a_0 + \sum_{t=1}^{T-1}(\beta_{t+1} - \beta_t)a_t - \beta_T a_T \le \beta_T R^2. \tag{27}$$

The uniform approximation bound therefore yields

$$F(\boldsymbol{y}_T) - F^\star \le \frac{\beta_T R^2}{2\Gamma_T} + c_p m \rho_T^p + \frac{c_p m}{\Gamma_T}\sum_{t=1}^{T} \Gamma_{t-1}(\rho_{t-1}^p - \rho_t^p). \tag{28}$$

The first term is

$$\frac{\beta_T R^2}{2\Gamma_T} = \frac{2\|\boldsymbol{A}\|^2 R^2 \bar{\rho}^{p-2}}{(T+1)^p}, \tag{29}$$

and the second is $c_p m \bar{\rho}^p (T+1)^{-p}$. For the drift term, the mean-value theorem gives

$$t^{-p} - (t+1)^{-p} \le p t^{-p-1}, \tag{30}$$

and $\Gamma_{t-1} \le t^2/2$. Since $1 \le p < 2$,

$$\sum_{t=1}^{T} t^{1-p} \le \frac{(T+1)^{2-p}}{2-p}. \tag{31}$$

Consequently,

$$\begin{aligned}\frac{c_p m}{\Gamma_T}\sum_{t=1}^{T} \Gamma_{t-1}(\rho_{t-1}^p - \rho_t^p) &\le \frac{c_p m p \bar{\rho}^p}{2\Gamma_T}\sum_{t=1}^{T} t^{1-p} \\ &\le \frac{m\bar{\rho}^p}{2}\frac{(T+1)^{1-p}}{T} \le m\bar{\rho}^p(T+1)^{-p}.\end{aligned} \tag{32}$$

Substitution into (28) gives the two-term bound

$$F(\boldsymbol{y}_T) - F^\star \le \frac{1}{(T+1)^p}\left[2\|\boldsymbol{A}\|^2 R^2 \bar{\rho}^{p-2} + \frac{p+2}{2p} m\bar{\rho}^p\right]. \tag{33}$$

It remains to optimize over $\bar{\rho}$. Differentiating the bracket in (33) gives

$$2\|\boldsymbol{A}\|^2 R^2 (p-2)\bar{\rho}^{p-3} + \frac{p+2}{2} m\bar{\rho}^{p-1} = 0, \tag{34}$$

whose unique positive solution is

$$\bar{\rho}^2 = \frac{4(2-p)}{(p+2)m}\|\boldsymbol{A}\|^2 R^2. \tag{35}$$

At this value, the bracket equals

$$\frac{4}{p}\left(\frac{p+2}{4(2-p)}\right)^{\frac{2-p}{2}} m^{\frac{2-p}{2}} (\|\boldsymbol{A}\|R)^p, \tag{36}$$

which proves (15). The oracle count is the same as in the fixed-smoothing variant. □

A.2.4 Proof of Theorem 3.4

The proof has two ingredients. First, we adapt the rotating hard-instance construction of Ouyang & Xu (2021) to a square linear system whose right-hand side is fixed before the hard matrix is selected. Second, we embed this Euclidean residual into an m-dimensional subspace on which the ℓ_p and Euclidean norms have the required dimension dependence. The embedding follows from the Gelfand-width estimate of Foucart et al. (2010).

A fixed-right-hand-side hard instance.

Lemma A.2. *Let $Q \geq 1$ be an integer, let $\Lambda, D > 0$, and set $d = 4Q + 9$. There exists a vector $\widehat{\boldsymbol{b}} \in \mathbb{R}^d$, depending only on (Q, Λ, D), with the following property. For every deterministic method that is given $\widehat{\boldsymbol{b}}$ and the scalar side information (Λ, D, Q) and makes at most Q adaptive queries to a paired oracle of the form*

$$(\boldsymbol{\xi}, \boldsymbol{\nu}) \mapsto (\widehat{\boldsymbol{A}}\boldsymbol{\xi}, \widehat{\boldsymbol{A}}^\top \boldsymbol{\nu}), \qquad \boldsymbol{\xi}, \boldsymbol{\nu} \in \mathbb{R}^d,$$

there exists an invertible matrix $\widehat{\boldsymbol{A}} \in \mathbb{R}^{d\times d}$ satisfying $\|\widehat{\boldsymbol{A}}\| = \Lambda$ such that the unique solution $\boldsymbol{\xi}^\star$ of $\widehat{\boldsymbol{A}}\boldsymbol{\xi} = \widehat{\boldsymbol{b}}$ satisfies $\|\boldsymbol{\xi}^\star\| = D$, while the method's output $\boldsymbol{\xi}_Q$ obeys

$$\|\widehat{\boldsymbol{A}}\boldsymbol{\xi}_Q - \widehat{\boldsymbol{b}}\| \geq \frac{\sqrt{3}}{4\sqrt{2}} \frac{\Lambda D}{2Q+5}. \tag{37}$$

Once (Q, Λ, D) are fixed, the vector $\widehat{\boldsymbol{b}}$ is fixed independently of both the method and $\widehat{\boldsymbol{A}}$, and is available before the first oracle query. Correspondingly, the regression right-hand side $\boldsymbol{b}$ in the final hard instance is fixed independently of $\boldsymbol{A}$.

Proof. Set

$$\ell := 2Q + 4, \qquad d := 2\ell + 1 = 4Q + 9. \tag{38}$$

Let $\boldsymbol{B} \in \mathbb{R}^{2\ell\times 2\ell}$ be the lower-bidiagonal matrix with

$$B_{ii} = 1 \quad (i = 1, \ldots, 2\ell), \qquad B_{i,i-1} = -1 \quad (i = 2, \ldots, 2\ell), \tag{39}$$

and all other entries equal to zero. Define

$$\boldsymbol{C} := \begin{pmatrix} \boldsymbol{B} & \boldsymbol{0} \\ \boldsymbol{0}^\top & 2 \end{pmatrix} \in \mathbb{R}^{d\times d}, \qquad \boldsymbol{c} := \begin{pmatrix} \boldsymbol{1}_{2\ell} \\ 0 \end{pmatrix} \in \mathbb{R}^d, \tag{40}$$

and

$$\boldsymbol{A}_0 := \frac{\Lambda}{2}\boldsymbol{C}, \qquad \boldsymbol{b}_0 := \frac{\Lambda}{2}\boldsymbol{c}. \tag{41}$$

For every $\boldsymbol{w} \in \mathbb{R}^{2\ell}$,

$$\|\boldsymbol{B}\boldsymbol{w}\|^2 = w_1^2 + \sum_{i=2}^{2\ell} (w_i - w_{i-1})^2 \leq 4\|\boldsymbol{w}\|^2, \tag{42}$$

so $\|\boldsymbol{B}\| \le 2$. The final scalar block shows $\|\boldsymbol{C}\| = 2$, and hence $\|\boldsymbol{A}_0\| = \Lambda$. Moreover, $\boldsymbol{C}$ is invertible, and the unique solution of $\boldsymbol{A}_0\boldsymbol{\xi} = \boldsymbol{b}_0$ is

$$\boldsymbol{\xi}_0^\star = (1, 2, \dots, 2\ell, 0)^\top. \tag{43}$$

Consequently,

$$\|\boldsymbol{\xi}_0^\star\| = \left(\frac{\ell(2\ell+1)(4\ell+1)}{3}\right)^{1/2}. \tag{44}$$

Set

$$a := \frac{D}{\|\boldsymbol{\xi}_0^\star\|}, \qquad \widehat{\boldsymbol{b}} := a\boldsymbol{b}_0 = \frac{\Lambda D}{2\|\boldsymbol{\xi}_0^\star\|}\begin{pmatrix}\mathbf{1}_{2\ell}\\ 0\end{pmatrix}. \tag{45}$$

This vector depends only on (Q, Λ, D), before any hard matrix is selected.

Fix an arbitrary deterministic paired-oracle method $\mathcal{N}$ for the scaled system with right-hand side $\widehat{\boldsymbol{b}}$. A method using fewer than Q calls may be padded with repeated zero queries, so it suffices to consider an output after exactly Q oracle rounds. We construct a method $\widehat{\mathcal{N}}$ in the first-order oracle model of Ouyang & Xu (2021). Whenever $\mathcal{N}$ requests $(\boldsymbol{\xi}, \boldsymbol{\nu})$, the method $\widehat{\mathcal{N}}$ queries the Ouyang–Xu oracle at $(\boldsymbol{\xi}/a, \boldsymbol{\nu})$. If the oracle returns

$$\big(\nabla f_{\mathrm{aux}}(\boldsymbol{\xi}/a), \boldsymbol{A}(\boldsymbol{\xi}/a), \boldsymbol{A}^\top\boldsymbol{\nu}\big),$$

then $\widehat{\mathcal{N}}$ discards the gradient and supplies

$$\big(a\boldsymbol{A}(\boldsymbol{\xi}/a), \boldsymbol{A}^\top\boldsymbol{\nu}\big) = \big(\boldsymbol{A}\boldsymbol{\xi}, \boldsymbol{A}^\top\boldsymbol{\nu}\big)$$

to $\mathcal{N}$. If $\mathcal{N}$ outputs $\boldsymbol{\xi}_Q$, then $\widehat{\mathcal{N}}$ outputs $\boldsymbol{\xi}_Q/a$. Thus $\widehat{\mathcal{N}}$ makes the same number of oracle calls and is a valid deterministic first-order method in their model.

Fix any positive smoothness parameter L_{aux} for the auxiliary objective f_{aux} in the Ouyang–Xu affine-constrained instance; its value is immaterial because $\widehat{\mathcal{N}}$ discards the gradient response. Apply the rotating hard-instance construction underlying the general-method lower bound of Ouyang & Xu (2021) with query index $t = Q$, internal parameter $\ell = 2Q + 4$, and square dimensions $d \times d$. Its dimensional requirement holds because

$$Q < \frac{d}{4} - 2 = Q + \frac{1}{4}. \tag{46}$$

The construction provides orthogonal matrices $\boldsymbol{U}, \boldsymbol{V}$ with $\boldsymbol{V}\boldsymbol{b}_0 = \boldsymbol{b}_0$ and a rotated matrix

$$\widehat{\boldsymbol{A}} := \boldsymbol{V}^\top\boldsymbol{A}_0\boldsymbol{U} \tag{47}$$

such that $\|\widehat{\boldsymbol{A}}\| = \Lambda$ and the output of $\widehat{\mathcal{N}}$ satisfies

$$\|\widehat{\boldsymbol{A}}(\boldsymbol{\xi}_Q/a) - \boldsymbol{b}_0\| \ge \frac{\sqrt{3}}{4\sqrt{2}}\frac{\Lambda\|\boldsymbol{\xi}_0^\star\|}{2Q+5}. \tag{48}$$

The matrix $\widehat{\boldsymbol{A}}$ is invertible because $\boldsymbol{A}_0$ is invertible and $\boldsymbol{U}, \boldsymbol{V}$ are orthogonal. Since $\boldsymbol{V}\boldsymbol{b}_0 = \boldsymbol{b}_0$, the unique solution of $\widehat{\boldsymbol{A}}\boldsymbol{\xi} = \boldsymbol{b}_0$ is $\widehat{\boldsymbol{\xi}}^\star = \boldsymbol{U}^\top\boldsymbol{\xi}_0^\star$, and $\|\widehat{\boldsymbol{\xi}}^\star\| = \|\boldsymbol{\xi}_0^\star\|$. Therefore, the unique solution of $\widehat{\boldsymbol{A}}\boldsymbol{\xi} = \widehat{\boldsymbol{b}}$ is

$$\boldsymbol{\xi}^\star := a\widehat{\boldsymbol{\xi}}^\star, \qquad \|\boldsymbol{\xi}^\star\| = a\|\boldsymbol{\xi}_0^\star\| = D. \tag{49}$$

Finally, multiplying (48) by a gives

$$\begin{aligned}\|\widehat{\boldsymbol{A}}\boldsymbol{\xi}_Q - \widehat{\boldsymbol{b}}\| &= a\|\widehat{\boldsymbol{A}}(\boldsymbol{\xi}_Q/a) - \boldsymbol{b}_0\| \\ &\ge \frac{\sqrt{3}}{4\sqrt{2}}\frac{\Lambda D}{2Q+5},\end{aligned} \tag{50}$$

which is (37). □

A Euclidean-section embedding.

Lemma A.3. *There exists a universal constant $c_0 \in (0,1)$ such that, for all integers $m, d \geq 1$ with $d \leq m/2$, there is a matrix $\boldsymbol{R} \in \mathbb{R}^{m\times d}$ satisfying $\boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}_d$ and, simultaneously for every $\boldsymbol{s} \in \mathbb{R}^d$ and every $p \in [1,2]$,*

$$\|\boldsymbol{R}\boldsymbol{s}\|_p^p \geq c_0{}^p m^{1-p/2}\|\boldsymbol{s}\|^p. \tag{51}$$

Proof. To avoid confusing the ambient dimension m with the width order, set

$$r := \lceil m/2 \rceil. \tag{52}$$

The upper estimate in Foucart et al. (2010, Theorem 1.1), applied with their exponents $p_{\mathrm{F}} = 1$, $q_{\mathrm{F}} = 2$, ambient dimension m, and width order r, gives a universal constant $C > 0$ such that

$$d_r(B_1^m, \ell_2^m) \leq C\left(\frac{1+\log(m/r)}{r}\right)^{1/2} \leq \frac{C_0}{\sqrt{m}} \tag{53}$$

for another universal constant $C_0 > 0$. By the definition of the Gelfand width, there exists a matrix $\boldsymbol{G} \in \mathbb{R}^{r\times m}$ such that, on $\mathcal{E} := \ker(\boldsymbol{G})$,

$$\|\boldsymbol{v}\|_1 \geq c_0\sqrt{m}\|\boldsymbol{v}\|, \qquad \boldsymbol{v} \in \mathcal{E}, \tag{54}$$

for a universal $c_0 \in (0,1)$. Moreover,

$$\dim(\mathcal{E}) \geq m - r = \lfloor m/2 \rfloor \geq d. \tag{55}$$

Choose a d-dimensional subspace $\mathcal{E}_d \subseteq \mathcal{E}$, let $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_d$ be an orthonormal basis of $\mathcal{E}_d$, and define

$$\boldsymbol{R} := [\boldsymbol{r}_1 \ \cdots \ \boldsymbol{r}_d]. \tag{56}$$

Then $\boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}_d$, and $\|\boldsymbol{R}\boldsymbol{s}\| = \|\boldsymbol{s}\|$ for every $\boldsymbol{s} \in \mathbb{R}^d$. For $p \in [1,2]$, the norm comparison $\|\boldsymbol{v}\|_1 \leq m^{1-1/p}\|\boldsymbol{v}\|_p$, together with (54), gives

$$\|\boldsymbol{R}\boldsymbol{s}\|_p \geq m^{1/p-1}\|\boldsymbol{R}\boldsymbol{s}\|_1 \geq c_0 m^{1/p-1/2}\|\boldsymbol{s}\|. \tag{57}$$

Taking the p-th power proves (51). The same subspace works simultaneously for every $p \in [1,2]$. □

Combining the two constructions. The first lemma gives the Euclidean residual lower bound, while the second transfers it to f with the required dependence on m.

Proof. Let $d = 4Q + 9$. Since $m \geq 2d$, choose $\boldsymbol{R} \in \mathbb{R}^{m\times d}$ from Lemma A.3. Since $n \geq d+1$, the orthogonal complement of $\boldsymbol{u}_0$ contains a d-dimensional subspace; choose $\boldsymbol{E} \in \mathbb{R}^{n\times d}$ satisfying

$$\boldsymbol{E}^\top \boldsymbol{E} = \boldsymbol{I}_d, \qquad \boldsymbol{E}^\top \boldsymbol{u}_0 = \boldsymbol{0}. \tag{58}$$

Let $\widehat{\boldsymbol{b}}$ be the vector from Lemma A.2, and set

$$\boldsymbol{b} := \boldsymbol{R}\widehat{\boldsymbol{b}}. \tag{59}$$

Both $\widehat{\boldsymbol{b}}$ and $\boldsymbol{b}$ are fixed before the hard matrix is selected.

Construct a deterministic d-dimensional method $\mathcal{N}$ from the given algorithm $\mathcal{M}$. The method $\mathcal{N}$ runs $\mathcal{M}$ with the prescribed dimensions and right-hand side $\boldsymbol{b}$. Whenever $\mathcal{M}$ queries $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^n \times \mathbb{R}^m$, the method $\mathcal{N}$ queries

$$(\boldsymbol{E}^\top \boldsymbol{u}, \boldsymbol{R}^\top \boldsymbol{v}) \in \mathbb{R}^d \times \mathbb{R}^d. \tag{60}$$

If the small oracle returns $(\boldsymbol{s}, \boldsymbol{t})$, then $\mathcal{N}$ returns $(\boldsymbol{R}\boldsymbol{s}, \boldsymbol{E}\boldsymbol{t})$ to $\mathcal{M}$. If $\mathcal{M}$ outputs $\boldsymbol{u}_Q$, then $\mathcal{N}$ outputs

$$\boldsymbol{\xi}_Q := \boldsymbol{E}^\top \boldsymbol{u}_Q. \tag{61}$$

This is a deterministic method making at most Q paired-oracle queries.

Apply Lemma A.2 to $\mathcal{N}$. It provides an invertible $\widehat{\boldsymbol{A}} \in \mathbb{R}^{d\times d}$, with $\|\widehat{\boldsymbol{A}}\| = \Lambda$, whose unique solution $\boldsymbol{\xi}^\star$ for right-hand side $\widehat{\boldsymbol{b}}$ satisfies $\|\boldsymbol{\xi}^\star\| = D$, and for which (37) holds. Define

$$\boldsymbol{A} := \boldsymbol{R}\widehat{\boldsymbol{A}}\boldsymbol{E}^\top. \tag{62}$$

For every query $(\boldsymbol{u}, \boldsymbol{v})$, the small oracle returns

$$\left(\widehat{\boldsymbol{A}}\boldsymbol{E}^\top \boldsymbol{u}, \widehat{\boldsymbol{A}}^\top \boldsymbol{R}^\top \boldsymbol{v}\right), \tag{63}$$

and the lifted responses are exactly

$$\left(\boldsymbol{R}\widehat{\boldsymbol{A}}\boldsymbol{E}^\top \boldsymbol{u}, \boldsymbol{E}\widehat{\boldsymbol{A}}^\top \boldsymbol{R}^\top \boldsymbol{v}\right) = (\boldsymbol{A}\boldsymbol{u}, \boldsymbol{A}^\top \boldsymbol{v}). \tag{64}$$

Thus the transcript simulated by $\mathcal{N}$ is precisely the transcript of $\mathcal{M}$ on the final instance $(\boldsymbol{A}, \boldsymbol{b})$.

We verify the instance properties. Since $\boldsymbol{R}$ and $\boldsymbol{E}$ have orthonormal columns,

$$\|\boldsymbol{A}\| \le \|\widehat{\boldsymbol{A}}\|. \tag{65}$$

Conversely, for every $\boldsymbol{\xi} \in \mathbb{R}^d$,

$$\|\boldsymbol{A}\boldsymbol{E}\boldsymbol{\xi}\| = \|\boldsymbol{R}\widehat{\boldsymbol{A}}\boldsymbol{\xi}\| = \|\widehat{\boldsymbol{A}}\boldsymbol{\xi}\|, \qquad \|\boldsymbol{E}\boldsymbol{\xi}\| = \|\boldsymbol{\xi}\|, \tag{66}$$

so taking the supremum over unit $\boldsymbol{\xi}$ gives $\|\boldsymbol{A}\| = \|\widehat{\boldsymbol{A}}\| = \Lambda$.

Define

$$\boldsymbol{u}^\star := \boldsymbol{u}_0 + \boldsymbol{E}\boldsymbol{\xi}^\star. \tag{67}$$

By (58),

$$\boldsymbol{A}\boldsymbol{u}^\star = \boldsymbol{R}\widehat{\boldsymbol{A}}\boldsymbol{\xi}^\star = \boldsymbol{R}\widehat{\boldsymbol{b}} = \boldsymbol{b}, \tag{68}$$

so $F(\boldsymbol{u}^\star) = 0$ and therefore $F^\star = 0$. If $\boldsymbol{u}$ is any minimizer, then $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b}$, and hence

$$\boldsymbol{R}\big(\widehat{\boldsymbol{A}}\boldsymbol{E}^\top \boldsymbol{u} - \widehat{\boldsymbol{b}}\big) = \boldsymbol{0}. \tag{69}$$

The matrix $\boldsymbol{R}$ has full column rank and $\widehat{\boldsymbol{A}}$ is invertible, so $\boldsymbol{E}^\top \boldsymbol{u} = \boldsymbol{\xi}^\star$. Consequently,

$$\|\boldsymbol{u} - \boldsymbol{u}_0\| \ge \|\boldsymbol{E}^\top(\boldsymbol{u} - \boldsymbol{u}_0)\| = \|\boldsymbol{\xi}^\star\| = D. \tag{70}$$

Equality is attained by $\boldsymbol{u}^\star$, proving $\operatorname{dist}(\boldsymbol{u}_0, \mathcal{U}^\star) = D$.

Finally, with $\boldsymbol{\xi}_Q = \boldsymbol{E}^\top \boldsymbol{u}_Q$,

$$\begin{aligned} F(\boldsymbol{u}_Q) - F^\star &= \frac{1}{p}\|\boldsymbol{R}\big(\widehat{\boldsymbol{A}}\boldsymbol{\xi}_Q - \widehat{\boldsymbol{b}}\big)\|_p^p \\ &\ge \frac{c_0^p}{p} m^{1-p/2} \|\widehat{\boldsymbol{A}}\boldsymbol{\xi}_Q - \widehat{\boldsymbol{b}}\|^p \\ &\ge \frac{1}{p}\left(\frac{\sqrt{3}\, c_0}{4\sqrt{2}}\right)^p m^{\frac{2-p}{2}} \left(\frac{\Lambda D}{2Q+5}\right)^p. \end{aligned} \tag{71}$$

This is (17). □

B Additional Discussions

B.1 Fixed smoothing via restart doubling

The fixed-smoothing guarantee in Theorem 3.2 uses a radius tuned to the terminal horizon. When the horizon is not known in advance, a restart-doubling scheme removes this requirement without changing the order of the rate. This construction runs the fixed-smoothing variant with horizon-tuned radii at geometrically increasing stage lengths. It is distinct from the variable-smoothing variant in Theorem 3.3, which follows a single evolving surrogate path.

Proposition B.1. *For stages $k = 0, 1, 2, \ldots$, set $T_k := 2^k$ and run the fixed-smoothing variant of Theorem 3.2 for T_k iterations from the common initial point $\mathbf{u}_0$, using the radius tuned to T_k. After any total budget $N \ge 1$, denote the output of the last completed stage by $\mathbf{u}_N$ and return it. Then*

$$F(\boldsymbol{u}_N) - F^\star \le \frac{8^p}{p}\, m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{N}\right)^p. \tag{72}$$

The resulting scheme requires neither a prescribed terminal horizon nor any bounded-iterate assumption.

Proof. Let K be the last completed stage. Since $\sum_{k=0}^{K} 2^k = 2^{K+1} - 1 \le N$, while stage $K+1$ is not completed, we have $N < 2^{K+2} - 1$, and therefore $T_K = 2^K \ge N/4$. Applying Theorem 3.2 to stage K gives

$$F(\boldsymbol{u}_N) - F^\star \le \frac{2^p}{p} m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{T_K} \right)^p \le \frac{8^p}{p} m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{N} \right)^p . \tag{73}$$

□

B.2 Constrained variable smoothing

A plain Euclidean projection is not compatible with a general nonsmooth ω. The correct bounded-domain variant uses the proximal map of the sum of ω and the ball indicator.

Corollary B.2 (Variable smoothing without a bounded-iterate assumption)**.** *Let $\mathcal{B}_D := \{\boldsymbol{u} : \|\boldsymbol{u} - \boldsymbol{u}_0\| \le D\}$, and assume that the constrained proximal map of $\omega + \iota_{\mathcal{B}_D}$ is available. Replace the $\boldsymbol{z}$-update in Algorithm 1 by*

$$\boldsymbol{z}_t := \operatorname{prox}_{(\omega + \iota_{\mathcal{B}_D})/\gamma_t} \left(\boldsymbol{z}_{t-1} - \gamma_t^{-1} \boldsymbol{g}_t \right). \tag{74}$$

Run the variable schedule of Theorem 3.3 with $R = 2D$. Then

$$F(\boldsymbol{y}_T) - F^\star \le 2^p C_p \, m^{\frac{2-p}{2}} \left(\frac{\|\boldsymbol{A}\| D}{T+1} \right)^p . \tag{75}$$

The method uses exactly $2T$ individual matrix–vector products and T evaluations of the constrained proximal map.

Proof. Choose a nearest minimizer $\boldsymbol{u}^\star \in \mathcal{U}^\star$. Then $\|\boldsymbol{u}^\star - \boldsymbol{u}_0\| \le D$, so $\boldsymbol{u}^\star \in \mathcal{B}_D$. The constrained problem

$$\min_{\boldsymbol{u}} \; f(\boldsymbol{u}) + \omega(\boldsymbol{u}) + \iota_{\mathcal{B}_D}(\boldsymbol{u})$$

has the same optimal value $F^\star$, because it contains the global minimizer $\boldsymbol{u}^\star$. Update (74) is exactly Algorithm 1 applied to the regularizer $\omega + \iota_{\mathcal{B}_D}$, so the composite three-point inequality in the proof of Lemma 3.1 remains valid.

Every $\boldsymbol{z}_t$ lies in $\mathcal{B}_D$, and every $\boldsymbol{y}_t$, being a convex combination of points in this ball, also lies there. Hence

$$\|\boldsymbol{z}_t - \boldsymbol{u}^\star\| \le \|\boldsymbol{z}_t - \boldsymbol{u}_0\| + \|\boldsymbol{u}_0 - \boldsymbol{u}^\star\| \le 2D.$$

Theorem 3.3 applies with $R = 2D$, and its objective gap is exactly $F(\boldsymbol{y}_T) - F^\star$, because $\boldsymbol{y}_T \in \mathcal{B}_D$. This proves (75); the constrained proximal evaluation requires no matrix–vector product. □

B.3 Backtracking line search

Proposition B.3. *Consider the radius schedule and bounded-sequence assumption of Theorem 3.3. At iteration t, accept a local estimate $\widehat{L}_t$ for which the test* (16) *holds, set $\gamma_t = \alpha_t \widehat{L}_t$, and impose*

$$0 < \widehat{L}_1 \le L_1, \qquad \frac{t^2 - 1}{t^2} \widehat{L}_{t-1} \le \widehat{L}_t \le L_t \quad (t \ge 2). \tag{76}$$

Then the conclusion (15) *remains valid. A backtracking search capped at L_t always terminates.*

Proof. The accepted test is precisely the smooth upper-model inequality needed in (18), evaluated at the actual pair $(\boldsymbol{x}_t, \boldsymbol{y}_t)$. All remaining steps in the proof of Lemma 3.1 use only convexity and proximal optimality, so the lemma remains valid with

$$\widehat{\beta}_t := \alpha_t^2 \widehat{L}_t \Gamma_t = \frac{2t}{t+1} \widehat{L}_t.$$

The lower safeguard in (76) is equivalent to $\widehat{\beta}_t \ge \widehat{\beta}_{t-1}$. Consequently, the Abel summation argument in the proof of Theorem 3.3 applies unchanged. Since $\widehat{L}_T \le L_T$, we also have $\widehat{\beta}_T \le \beta_T$, so the same worst-case bound follows.

The global constant L_t satisfies the test by Proposition 2.5. Moreover, for the variable schedule, $L_t > L_{t-1}$, while the lower safeguard is strictly smaller than $\widehat{L}_{t-1} \le L_{t-1}$. Thus the admissible interval in (76) contains L_t, and a search capped at that value terminates. □

Each line-search trial evaluates one candidate proximal point and $f_t(\boldsymbol{y}_t)$; the latter requires one additional forward product. Rejected trials therefore add one constrained or unconstrained proximal evaluation and one forward product each, but no additional adjoint product. Without the lower safeguard in (76), every accepted step still has a certified local upper model, but the global Abel summation argument need not hold.

B.4 Least-squares initialization and residual-based scale selection

For $\omega \equiv 0$, we compute a least-squares solution and residual

$$\boldsymbol{u}_{\mathrm{OLS}} \in \arg\min_{\boldsymbol{u}} \tfrac{1}{2}\|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_2^2, \qquad \boldsymbol{r}_{\mathrm{OLS}} = \boldsymbol{A}\boldsymbol{u}_{\mathrm{OLS}} - \boldsymbol{b}. \tag{77}$$

For nonzero residual, we initialize Algorithm 1 at $\boldsymbol{u}_{\mathrm{OLS}}$ and use

$$\bar{\rho}_{\mathrm{OLS}} = 2\|\boldsymbol{r}_{\mathrm{OLS}}\|_\infty, \qquad \rho_t = \frac{\bar{\rho}_{\mathrm{OLS}}}{t+1}, \qquad L_t = \|\boldsymbol{A}\|^2 \rho_t^{p-2}, \qquad \gamma_t = \alpha_t L_t. \tag{78}$$

There is no backtracking. In exact arithmetic, a zero residual already solves the unregularized problem.

The factor two makes $\rho_1 = \|\boldsymbol{r}_{\mathrm{OLS}}\|_\infty$, so all initial residual coordinates lie in the quadratic part of the first surrogate. For an exact least-squares solution, the normal equations give

$$\nabla f_{\rho_1}(\boldsymbol{u}_{\mathrm{OLS}}) = \rho_1^{p-2}\boldsymbol{A}^\top \boldsymbol{r}_{\mathrm{OLS}} = \boldsymbol{0}. \tag{79}$$

Convexity therefore makes $\boldsymbol{u}_{\mathrm{OLS}}$ a minimizer of the first surrogate, providing a quadratic starting point for continuation (Bubeck et al., 2018).

The two-term estimate in (33) holds for any positive $\bar{\rho}$ before its final optimization in the proof of Theorem 3.3. Substituting $\bar{\rho} = \bar{\rho}_{\mathrm{OLS}}$ gives an $\mathcal{O}(T^{-p})$ objective-gap rate if the accelerated iterates admit a common bound $\|\boldsymbol{z}_t - \boldsymbol{u}^\star\| \le R$ independent of T. The residual rule does not establish this boundedness and need not attain the optimally balanced constant. Neither R nor a reference solution is needed to run the method. An approximate least-squares solution retains this conditional rate for a positive residual scale; an exact least-squares solution is required only for the stationarity identity above.

C Experimental details

C.1 Algorithm settings and evaluation

Smoothing methods. The fixed-smoothing, variable-smoothing, and backtracking methods apply Algorithm 1. In the two unregularized studies, they start from $\boldsymbol{u}_0 = \boldsymbol{0}$ with $\omega \equiv 0$. In the regularized and constrained studies, they retain ω through its proximal operator. Fixed smoothing uses the radius in (12), and variable smoothing uses the schedule in (14). In the controlled study, fixed smoothing uses $T = 30{,}000$ and the known initial distance D, while variable smoothing and backtracking use $R = D$ as a distance proxy. The initial points, horizons, and distance choices for the other studies are specified in their respective setups. The least-squares initialization variant used in the real-data study is described in Appendix C.3.3.

Backtracking uses the variable-smoothing schedule and tests (16) with a local curvature estimate. The first trial is $L_1/2$; for $t \ge 2$, it starts from

$$\min\left\{L_t, \frac{1}{2}\widehat{L}_{t-1}\left(\frac{\rho_t}{\rho_{t-1}}\right)^{p-2}\right\}.$$

A failed trial doubles the estimate, capped at L_t. This implementation omits the lower bound in (76) required by Proposition B.3.

Baseline methods. UFGM follows the Euclidean method of Nesterov (2015). The unregularized studies use $\omega \equiv 0$, while the regularized and constrained studies include the proximal operator of ω. The initial curvature trial is $L_0 = 1$, and failed model tests double the trial estimate.

GAGD+ follows Diakonikolas & Guzmán (2024). The unregularized studies use the non-composite ℓ_p-regression specialization with its native ℓ_p geometry and no supplied induced operator norm. The regularized and constrained studies use the Euclidean specialization with zero uniform-convexity parameter and cumulative regularization weights. The initial curvature trial is $M_0 = 1$, and failed model tests double the trial estimate.

For UFGM and GAGD+, the prescribed accuracy enters the recurrence and is set to $\epsilon = \tau(F(\boldsymbol{u}_0) - F_{\mathrm{ref}})$ before each run. The controlled study uses $\tau = 10^{-9}$, with both reported thresholds evaluated along that run. The real-data, regularized, and constrained studies use separate runs for $\tau = 10^{-4}$ and $\tau = 10^{-6}$, and each target is evaluated using its matching configuration.

AGDA uses the unconstrained, target-free method of Ren et al. (2025) in the two unregularized studies, with $\bar{r}_0 = 10^{-2}$ and $\beta_0 = 10^{-3}$, and returns the iterate with the lowest objective value so far. With iterations indexed from $k = 0$, the implementation uses the bisection tolerance $\beta_0/[2(k+1)^2]$ in place of $\beta_0/(2k^2)$ in their Algorithm 5. Bisection retains the upper endpoint satisfying the line-search condition and uses a machine-precision lower bound on the interval-width tolerance. Baseline initialization parameters are held fixed across instances.

Computational cost and evaluation. Each multiplication by $\boldsymbol{A}$ or $\boldsymbol{A}^\top$ counts as one product. The count includes initialization and all line-search trials, including trials interrupted by the budget limit. Products used solely for evaluation and reporting are recorded separately and excluded. A previously computed product can be reused exactly without increasing the count. For fixed and variable smoothing without backtracking, each completed iteration uses one forward and one adjoint product; a completed backtracking iteration with N_t^{rej} rejected trials uses $3 + N_t^{\mathrm{rej}}$ products.

Evaluation uses the original nonsmoothed objective at each method's prescribed output after every completed iteration. The cost of reaching a target is the cumulative product count at the first output satisfying it. The main comparisons in the two unregularized studies use a budget of $60{,}000$ products; Appendix C.3.5 examines a larger budget on selected real-data problems. The regularized and constrained budgets are specified with each study. UFGM and GAGD+ stop at their configured target or the budget limit. The other methods are evaluated at both reported thresholds along a single run. A target not reached within the budget is treated as a non-hit.

C.2 Controlled studies

C.2.1 Setup

The controlled study considers the unregularized objective $F(\boldsymbol{u}) = \|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_p^p/p$ with $1 < p < 2$.

Synthetic instance generation. The instances have a prescribed spectral norm, condition number, initial distance, and optimal residual scale. Their minimizers and optimal values are known by construction. Algorithm 2 gives the generation procedure. The matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ has condition number κ and spectral norm $\|\boldsymbol{A}\| = 1$. Let $\boldsymbol{U} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{n\times n}$ have orthonormal columns obtained from independent Gaussian matrices, and set

$$\boldsymbol{A} = \boldsymbol{U}\,\mathrm{Diag}(\sigma_1, \ldots, \sigma_n)\boldsymbol{V}^\top, \qquad \sigma_j = \kappa^{-\frac{j-1}{n-1}}, \quad j = 1, \ldots, n. \tag{80}$$

The singular values are therefore logarithmically spaced between 1 and $1/\kappa$, which gives $\|\boldsymbol{A}\| = 1$ and $\kappa(\boldsymbol{A}) = \kappa$.

The initial point is $\boldsymbol{u}_0 := \boldsymbol{0} \in \mathbb{R}^n$, and the prescribed minimizer is $\boldsymbol{u}^\star = D\boldsymbol{d}/\|\boldsymbol{d}\|$, where $\boldsymbol{d} \in \mathbb{R}^n$ is drawn from the standard normal distribution. The initial distance D is constant across all instances.

The nonzero optimal residual is constructed from a direction in the null space of $\boldsymbol{A}^\top$. For a standard Gaussian vector $\boldsymbol{g} \in \mathbb{R}^m$, set

$$\boldsymbol{\theta} = \frac{(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^\top)\boldsymbol{g}}{\|(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^\top)\boldsymbol{g}\|}. \tag{81}$$

Define $(r_0)_i = \mathrm{sign}(\theta_i)|\theta_i|^{1/(p-1)}$ for each i, and set

$$\boldsymbol{r}^\star = \frac{\sqrt{m}\rho_{\mathrm{res}}}{\|\boldsymbol{r}_0\|}\boldsymbol{r}_0, \qquad \boldsymbol{b} = \boldsymbol{A}\boldsymbol{u}^\star - \boldsymbol{r}^\star, \qquad \rho_{\mathrm{res}} = 0.2. \tag{82}$$

Algorithm 2: Controlled instance generation for one paired seed

Input: base random seed s

Initialize: $m \leftarrow 200$, $n \leftarrow 50$, $D \leftarrow 10$, $\rho_{\mathrm{res}} \leftarrow 0.2$, $\boldsymbol{u}_0 \leftarrow \mathbf{0}$;
$p \in \{1.1, 1.25, 1.5, 1.75, 1.9\}$, $\kappa \in \{1, 10, 10^2, 10^3, 10^4\}$

Using seed s, draw independent Gaussian matrices and compute canonical QR factorizations to obtain $\boldsymbol{U} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{n\times n}$ with orthonormal columns

Draw $\boldsymbol{d} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_n)$ and set $\boldsymbol{u}^\star \leftarrow D\,\boldsymbol{d}/\|\boldsymbol{d}\|$

Draw $\boldsymbol{g} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_m)$ and set $\boldsymbol{\theta} \leftarrow (\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^\top)\boldsymbol{g}/\|(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^\top)\boldsymbol{g}\|$

for $p \in \{1.1, 1.25, 1.5, 1.75, 1.9\}$ **do**

  for $i = 1, \ldots, m$ **do**

    $(r_0)_i \leftarrow \mathrm{sign}(\theta_i)\,|\theta_i|^{1/(p-1)}$

  Rescale the residual to the prescribed RMS magnitude: $\boldsymbol{r}^\star \leftarrow \frac{\sqrt{m}\,\rho_{\mathrm{res}}}{\|\boldsymbol{r}_0\|}\,\boldsymbol{r}_0$

  Compute the exact reference value $F_{\mathrm{ref}} \leftarrow \frac{1}{p}\|\boldsymbol{r}^\star\|_p^p$

  for $\kappa \in \{1, 10, 10^2, 10^3, 10^4\}$ **do**

    for $j = 1, \ldots, n$ **do**

      $\sigma_j \leftarrow \kappa^{-(j-1)/(n-1)}$

    Construct $\boldsymbol{A} \leftarrow \boldsymbol{U}\ \mathrm{Diag}(\sigma_1, \ldots, \sigma_n)\,\boldsymbol{V}^\top$

    Set $\boldsymbol{b} \leftarrow \boldsymbol{A}\boldsymbol{u}^\star - \boldsymbol{r}^\star$

    Verify the spectral norm, condition number, initial distance, residual RMS, stationarity, and primal–dual agreement numerically

Output: the 25 paired controlled instances associated with seed s

The vector $\mathrm{sign}(\boldsymbol{r}^\star) \odot |\boldsymbol{r}^\star|^{p-1}$ is a positive multiple of $\boldsymbol{\theta}$, so $\boldsymbol{A}^\top\boldsymbol{\theta} = 0$ gives $\nabla F(\boldsymbol{u}^\star) = \boldsymbol{A}^\top\big(\mathrm{sign}(\boldsymbol{r}^\star) \odot |\boldsymbol{r}^\star|^{p-1}\big) = 0$. Convexity therefore establishes $\boldsymbol{u}^\star$ as a minimizer. The residual root-mean-square magnitude satisfies $\|\boldsymbol{r}^\star\|/\sqrt{m} = \rho_{\mathrm{res}}$ for every instance, and the optimal value is

$$F_{\mathrm{ref}} = F^\star = \frac{1}{p}\|\boldsymbol{r}^\star\|_p^p. \tag{83}$$

For each seed, the same orthogonal factors $(\boldsymbol{U}, \boldsymbol{V})$, optimizer direction, and projected residual direction are reused across all (p, κ) pairs. Different seeds use independent random draws. Within each seed and at fixed p, changing κ varies the singular values while holding the orthogonal factors, minimizer, and optimal residual fixed. Changing p modifies the loss exponent and the corresponding optimal residual.

Curve aggregation and target attainment. Convergence curves use a common grid of 200 matrix–vector product budgets: the integers 1 through 20, followed by 180 rounded logarithmically spaced budgets from 21 to 60,000. At each budget, each run contributes its last prescribed output whose cost does not exceed that budget. Runs that terminate earlier retain their terminal values at subsequent budgets, so all medians and interquartile ranges use the ten paired seeds. Quantiles follow the Hyndman–Fan type-7 convention. Aggregation uses these outputs directly, without interpolation or an additional running minimum. The plotted lines connect the median values, and markers show the first downward crossings of these lines at selected gap levels. Target-attainment costs use the first prescribed output satisfying each target in the full output sequence, independently of the plotting grid and the displayed vertical range of 10^{-9} to 2.

C.2.2 Detailed results

Table 3 summarizes target attainment over the full grid, and Table 4 reports the results for each (p, κ) pair. Median first-hit costs are reported together with attainment counts because they are computed only over runs that reach the target, and these subsets may differ across methods.

Variable smoothing achieves a lower terminal gap than each of UFGM, AGDA, and GAGD+ on all 60 paired instances with $p \le 1.25$ and $\kappa \ge 100$. At the common product budget, the median of

the per-instance ratio of the baseline terminal gap to the variable-smoothing terminal gap is 21.8 for UFGM, 134.5 for AGDA, and 103.1 for GAGD+.

Table 3: Target attainment within a budget of $60{,}000$ matrix–vector products. The two overall columns report the number of runs reaching each target across all 250 instances. Parentheses give the median first-hit cost among those runs. Each remaining column reports the number of runs reaching 10^{-9} across the 50 instances with a fixed p or κ. The largest count in each column is bold. Table 4 gives the results for individual (p, κ) pairs.

	Overall hits (median first-hit matvecs)	
Method	10^{-6}	10^{-9}
HS-APG-F	188 (4,016)	92 (3,618)
HS-APG-V	**201 (4,582)**	107 (9,436)
HS-APG-V-BT	197 (4,074)	113 (10,825)
AGDA	164 (5,258)	103 (11,494)
UFGM	193 (3,243)	**128 (5,064)**
GAGD+	187 (3,456)	108 (6,738)

	10^{-9} hits by exponent p				
Method	1.1	1.25	1.5	1.75	1.9
HS-APG-F	0	0	18	33	**41**
HS-APG-V	0	0	30	36	**41**
HS-APG-V-BT	0	0	**31**	**41**	**41**
AGDA	0	0	22	40	**41**
UFGM	**20**	**7**	28	37	36
GAGD+	5	0	22	40	**41**

	10^{-9} hits by $\kappa(\boldsymbol{A})$				
Method	1	10	10^2	10^3	10^4
HS-APG-F	30	28	20	13	1
HS-APG-V	30	30	**30**	16	1
HS-APG-V-BT	30	29	28	**24**	**2**
AGDA	30	28	24	20	1
UFGM	**47**	**40**	28	13	0
GAGD+	35	30	22	20	1

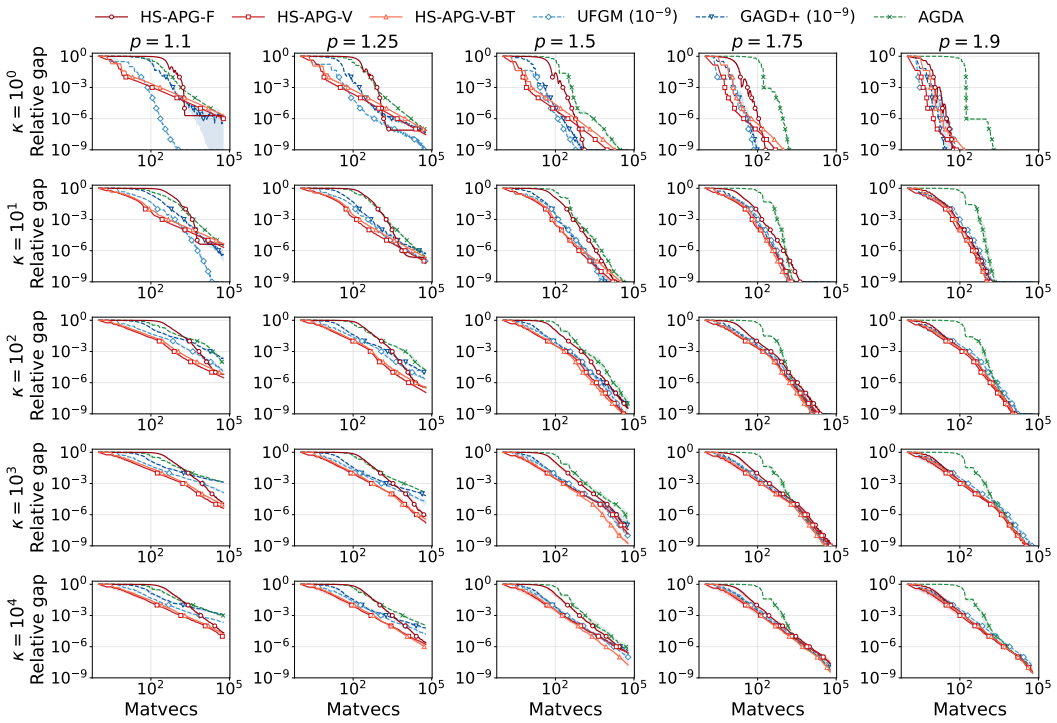

Figure 2: Complete controlled convergence grid over all five exponents and five condition numbers. Curves show median relative gaps over ten paired seeds against the number of matrix–vector products. Shaded bands span the 25th to 75th percentiles. UFGM and GAGD+ use target 10^{-9}, as in Figure 1.

Table 4: Target attainment for each (p, κ) pair within a budget of $60{,}000$ matrix–vector products. Each method entry gives the number of runs reaching the target out of ten paired seeds, followed by the median first-hit cost over those runs. Costs are measured in matrix–vector products; 0/– indicates that no run reaches the target. UFGM and GAGD+ are configured for target 10^{-9}, while the other methods do not require an accuracy target.

p	κ	Target	F	V	V-BT	AGDA	UFGM	GAGD+
1.1	1	10^{-6}	0/–	8/57,687	0/–	0/–	10/237	8/4,753.5
1.1	1	10^{-9}	0/–	0/–	0/–	0/–	10/1,096.5	5/12,801
1.1	10	10^{-6}	0/–	0/–	0/–	0/–	10/5,434.5	10/27,205.5
1.1	10	10^{-9}	0/–	0/–	0/–	0/–	10/20,332.5	0/–
1.1	10^2	10^{-6}	0/–	0/–	0/–	0/–	0/–	0/–
1.1	10^2	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.1	10^3	10^{-6}	0/–	0/–	0/–	0/–	0/–	0/–
1.1	10^3	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.1	10^4	10^{-6}	0/–	0/–	0/–	0/–	0/–	0/–
1.1	10^4	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.25	1	10^{-6}	10/1,391	10/4,797	10/9,593	10/16,126	10/550.5	10/2,439
1.25	1	10^{-9}	0/–	0/–	0/–	0/–	7/42,237	0/–
1.25	10	10^{-6}	10/6,508	10/9,583	10/19,153.5	10/32,451.5	10/7,606.5	10/22,171.5
1.25	10	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.25	10^2	10^{-6}	10/22,601	10/13,096	10/24,017	0/–	3/49,131	0/–
1.25	10^2	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.25	10^3	10^{-6}	8/44,925	10/25,446	10/31,052	0/–	0/–	0/–
1.25	10^3	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.25	10^4	10^{-6}	0/–	3/44,134	7/46,057	0/–	0/–	0/–
1.25	10^4	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–
1.5	1	10^{-6}	10/588	10/302	10/603	10/2,908.5	10/124.5	10/186
1.5	1	10^{-9}	10/1,256	10/9,410	10/18,816	10/30,009.5	10/580.5	10/960
1.5	10	10^{-6}	10/3,416	10/741	10/1,110	10/5,830.5	10/1,129.5	10/1,197
1.5	10	10^{-9}	8/15,521	10/15,560	9/30,240	8/36,559	10/5,431.5	10/7,426.5
1.5	10^2	10^{-6}	10/9,017	10/3,604	10/2,457	10/16,241	10/4,914	10/6,732
1.5	10^2	10^{-9}	0/–	10/38,734	8/37,333.5	4/39,737.5	8/37,965	2/26,748
1.5	10^3	10^{-6}	10/21,309	10/15,033	10/6,272	6/40,467	10/12,583.5	10/20,395.5
1.5	10^3	10^{-9}	0/–	0/–	4/53,079	0/–	0/–	0/–
1.5	10^4	10^{-6}	10/27,491	10/21,173	10/8,664	8/32,066.5	10/16,818	9/29,997
1.5	10^4	10^{-9}	0/–	0/–	0/–	0/–	0/–	0/–

Table 4 (continued).

p	κ	Target	F	V	V-BT	AGDA	UFGM	GAGD+
1.75	1	10^{-6}	10/108	10/30	10/60	10/926	10/36	10/42
1.75	1	10^{-9}	10/201	10/479	10/956.5	10/1,596.5	10/72	10/94.5
1.75	10	10^{-6}	10/947	10/408	10/360	10/1,117.5	10/550.5	10/475.5
1.75	10	10^{-9}	10/3,467	10/1,790	10/1,584	10/2,269	10/1,479	10/1,459.5
1.75	10^2	10^{-6}	10/3,348	10/1,977	10/1,630.5	10/2,521	10/2,497.5	10/1,897.5
1.75	10^2	10^{-9}	10/22,753	10/15,310	10/10,081	10/11,624	10/14,859	10/13,275
1.75	10^3	10^{-6}	10/8,101	10/6,454	10/4,415.5	10/5,752	10/6,673.5	10/5,172
1.75	10^3	10^{-9}	3/53,882	6/48,726	10/34,238	10/39,497	7/41,529	10/40,213.5
1.75	10^4	10^{-6}	10/9,523	10/7,675	10/5,071	10/6,629	10/7,653	10/6,034.5
1.75	10^4	10^{-9}	0/–	0/–	1/52,187	0/–	0/–	0/–
1.9	1	10^{-6}	10/32	10/10	10/20	10/172	10/30	10/18
1.9	1	10^{-9}	10/53	10/74	10/148	10/1,970	10/66	10/27
1.9	10	10^{-6}	10/401	10/300	10/345	10/1,072	10/549	10/393
1.9	10	10^{-9}	10/1,384	10/1,051	10/995	10/2,144.5	10/1,560	10/1,134
1.9	10^2	10^{-6}	10/1,737	10/1,334	10/1,772.5	10/2,458.5	10/2,662.5	10/1,678.5
1.9	10^2	10^{-9}	10/9,771	10/8,891	10/9,518.5	10/11,119	10/15,151.5	10/10,203
1.9	10^3	10^{-6}	10/4,273	10/3,891	10/4,364.5	10/5,376	10/6,879	10/4,479
1.9	10^3	10^{-9}	10/33,957	10/30,345	10/33,276.5	10/35,409.5	6/46,504.5	10/35,482.5
1.9	10^4	10^{-6}	10/4,998	10/4,574	10/5,058.5	10/6,476	10/7,944	10/5,194.5
1.9	10^4	10^{-9}	1/52,284	1/54,470	1/50,542	1/58,525	0/–	1/54,708

C.3 Real-data evaluations

C.3.1 Datasets and preprocessing

We construct regression problems from six UCI datasets: Power, Year, Protein, KEGG, CT-Slices, and Buzz (Hebrail & Berard, 2006; Bertin-Mahieux, 2011; Rana, 2013; Naeem & Asghar, 2011; Graf et al., 2011; Kawala et al., 2013). Table 5 lists the response variables and the resulting matrix dimensions. We exclude Date and Time from Power, the textual Pathway field from KEGG, and `patientId` from CT-Slices. Protein uses the nine physicochemical variables as predictors. Year uses the first $463{,}715$ observations in the official ordered training split; the final $51{,}630$ test observations are excluded. For Buzz, we use `Twitter.data` from the Twitter regression subset, with the rightmost column as the response.

Rows with missing values in any selected predictor or response are removed before preprocessing statistics are computed. For Power, both `?` and empty numerical fields are treated as missing. For CT-Slices, the value -0.25, which denotes histogram bins outside the image, is retained. Each predictor and the response are centered and scaled by their population root-mean-square deviations. Constant predictors are removed, and numerical redundancy is assessed by column-pivoted QR without using the response. The tolerance accounts for matrix dimensions and floating-point precision, with a separate tolerance for KEGG to account for the decimal precision of its source data. This procedure removes two redundant predictors from KEGG and five constant and six redundant predictors from CT-Slices. No predictors are removed from the other four datasets. All methods use the same retained predictor sets.

An intercept is then appended, and the augmented design matrix is divided by its spectral norm so that $\|\boldsymbol{A}\|_2 = 1$. This final scaling is not applied to $\boldsymbol{b}$. No logarithmic transformation or row normalization is used. Table 5 also reports the condition number $\kappa_2(\boldsymbol{A}) = \sigma_{\max}(\boldsymbol{A})/\sigma_{\min}(\boldsymbol{A})$, computed from the processed matrix used by all methods.

Table 5: Processed real-data optimization designs. The column count n includes the intercept.

Dataset	Response	m	n	$\kappa_2(\boldsymbol{A})$
Power	Global active power	2,049,280	7	3.8417
YearPredictionMSD	Release year	463,715	91	13.260
Protein	RMSD	45,730	10	67.626
KEGG	Edge count	64,608	26	100.90
CT-Slices	Axial location	53,500	374	583.46
Buzz/Twitter	Mean active discussions	583,250	78	7278.3

C.3.2 High-precision reference construction

For each dataset and exponent $p \in (1, 2)$, let

$$\boldsymbol{u}^\star \in \arg\min_{\boldsymbol{u}} F_p(\boldsymbol{u}), \qquad F_p^\star = F_p(\boldsymbol{u}^\star) = \min_{\boldsymbol{u}} \left\{ \frac{1}{p} \|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_p^p \right\}. \tag{84}$$

We compute a primal candidate $\boldsymbol{u}_{\mathrm{ref}}$ using BFGS with Hager–Zhang line search in 512-bit arithmetic (Nocedal & Wright, 2006; Hager & Zhang, 2005). Its objective value gives the upper bound

$$F_p^{\mathrm{U}} := F_p(\boldsymbol{u}_{\mathrm{ref}}) \geq F_p^\star. \tag{85}$$

We assess reference accuracy through a primal–dual interval (Dünner et al., 2016). With a lower bound $F_p^{\mathrm{L}} \leq F_p^\star$ and the common normalization point $\boldsymbol{u}_0 = \boldsymbol{0}$, we require

$$\frac{F_p^{\mathrm{U}} - F_p^{\mathrm{L}}}{F_p(\boldsymbol{u}_0) - F_p^{\mathrm{L}}} \leq 10^{-10}. \tag{86}$$

Objective gaps and stopping targets use $F_{\mathrm{ref}} = F_p^{\mathrm{L}}$. The primal candidate $\boldsymbol{u}_{\mathrm{ref}}$ is used only for diagnostics and distance-based parameters.

Let $q = p/(p-1)$. The Fenchel dual is

$$\max_{\boldsymbol{\lambda}:\, \boldsymbol{A}^\top \boldsymbol{\lambda} = 0} \left\{ D(\boldsymbol{\lambda}) := \boldsymbol{\lambda}^\top \boldsymbol{b} - \frac{1}{q} \|\boldsymbol{\lambda}\|_q^q \right\}. \tag{87}$$

The primal residual gives the direction

$$\boldsymbol{\lambda}_0 := -\operatorname{sign}(\boldsymbol{r}_{\mathrm{ref}}) \odot |\boldsymbol{r}_{\mathrm{ref}}|^{p-1}, \qquad \boldsymbol{r}_{\mathrm{ref}} := \boldsymbol{A}\boldsymbol{u}_{\mathrm{ref}} - \boldsymbol{b}, \tag{88}$$

which satisfies $\boldsymbol{A}^\top \boldsymbol{\lambda}_0 = -\nabla F_p(\boldsymbol{u}_{\mathrm{ref}})$. We project this direction onto $\ker(\boldsymbol{A}^\top)$:

$$\widehat{\boldsymbol{\lambda}} := \boldsymbol{\lambda}_0 - \boldsymbol{A}(\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{\lambda}_0, \tag{89}$$

with the linear systems solved using a high-precision Gram factorization.

Numerical projection can leave a nonzero feasibility residual $\boldsymbol{A}^\top \widehat{\boldsymbol{\lambda}}$. To account for it, let $0 < \underline{\sigma}(\boldsymbol{A}) \leq \sigma_{\min}(\boldsymbol{A})$ be a certified lower bound and set $R_{\boldsymbol{u}} = 2\|\boldsymbol{b}\|_p / \underline{\sigma}(\boldsymbol{A})$. Since $F_p(\boldsymbol{u}^\star) \leq F_p(\boldsymbol{0})$, we have $\|\boldsymbol{A}\boldsymbol{u}^\star - \boldsymbol{b}\|_p \leq \|\boldsymbol{b}\|_p$, which implies $\|\boldsymbol{u}^\star\|_2 \leq R_{\boldsymbol{u}}$. Fenchel's inequality then gives the corrected lower bound

$$F_p^{\mathrm{L}} := \max_{\alpha \geq 0} \left\{ \alpha \left(\widehat{\boldsymbol{\lambda}}^\top \boldsymbol{b} - R_{\boldsymbol{u}} \|\boldsymbol{A}^\top \widehat{\boldsymbol{\lambda}}\|_2 \right) - \frac{\alpha^q}{q} \|\widehat{\boldsymbol{\lambda}}\|_q^q \right\}. \tag{90}$$

Thus $F_p^{\mathrm{L}} \leq F_p^\star \leq F_p^{\mathrm{U}}$.

This residual-based construction meets the accuracy requirement for 17 of the 18 problems. For CT-Slices at $p = 1.1$, we instead maximize $D(Pz)$ using projected L-BFGS with Hager–Zhang line search, where

$$P := I - \boldsymbol{A}(\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top. \tag{91}$$

The primal candidate remains fixed. The resulting direction is reprojected, and the same corrected lower bound is evaluated in 512-bit arithmetic. This reduces the normalized interval width from approximately 1.08×10^{-6} to 2.16×10^{-11}. An independent 768-bit computation also satisfies the accuracy requirement, using reversed summation order and the weaker singular-value bound $17/10000$. All 18 reference intervals satisfy the 10^{-10} requirement.

C.3.3 ALGORITHM SETTINGS AND EVALUATION

Reference-distance setting. Fixed smoothing, variable smoothing, and variable smoothing with backtracking start from $\boldsymbol{u}_0 = \mathbf{0}$ and use $D_{\mathrm{ref}} = \|\boldsymbol{u}_{\mathrm{ref}}\|_2$ as a distance proxy. Fixed smoothing sets $D = D_{\mathrm{ref}}$, while both variable-smoothing variants set $R = D_{\mathrm{ref}}$ in their radius formulas. The primal–dual certificate bounds the objective error at $\boldsymbol{u}_{\mathrm{ref}}$; it does not certify D_{ref} as an upper bound on the initial or subsequent distances to a minimizer.

Least-squares initialization. Variable smoothing with least-squares initialization follows Appendix B.4. The least-squares point is computed from zero by unpreconditioned LSQR, using only products with $\boldsymbol{A}$ and $\boldsymbol{A}^\top$. LSQR stops when an explicitly computed normal residual satisfies

$$\frac{\|\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{u}_{\mathrm{LS}} - \boldsymbol{b})\|_2}{\|\boldsymbol{A}^\top \boldsymbol{b}\|_2} \leq 10^{-10}.$$

This residual is checked every ten iterations, when the recurrence indicates convergence or breakdown, and near the budget limit if sufficient products remain.

All products required for initialization count toward the shared budget, including the initial adjoint product and the residual checks. Initialization is performed separately for each problem, with no reuse across exponents. All 18 initializations satisfy the stopping criterion within the budget. The resulting least-squares residuals are nonzero in all 18 problems and determine the smoothing scale used in the subsequent variable-smoothing iterations, without reference-distance information.

Target attainment is evaluated at every completed LSQR iteration and every subsequent accelerated output, using the common denominator $F(\mathbf{0}) - F_{\mathrm{ref}}$. If a target is reached during initialization, its attainment cost is the cumulative cost at that iterate. As for the other methods, products used only to evaluate objectives for reporting are counted separately and excluded from the algorithmic budget. These products are not reused in the LSQR stopping test or the smoothing scale.

Evaluation and cost accounting. Variable smoothing with least-squares initialization begins with LSQR at zero; all other methods begin optimization at zero. Each run has a budget of $60{,}000$ matrix–vector products, including initialization and all line-search products. Product counts and objective evaluations follow Appendix C.1.

Target attainment is evaluated at relative gaps 10^{-4} and 10^{-6}, using the common denominator $F(\mathbf{0}) - F_{\mathrm{ref}}$. UFGM and GAGD+ are run separately for each target, and each target is evaluated using its matching configuration. The remaining methods are evaluated at both targets along a single run.

The four smoothing variants also have a $30{,}000$-iteration limit. Fixed smoothing uses this full horizon to set its radius in (12), and its first-hit costs at both targets are obtained from the same run.

C.3.4 DETAILED RESULTS AND ADAPTIVE RESTART

Target attainment. Tables 6 and 7 report first-hit costs at the two relative-gap targets.

Table 6: First-hit matrix–vector products for the 18 real-data problems at relative-gap target 10^{-4}.

Dataset	p	HS-APG-F	HS-APG-V	HS-APG-V-BT	GAGD+	UFGM	AGDA	HS-APG-OLS
Power	1.1	3,440	**38**	108	483	165	785	77
	1.5	352	36	44	54	129	646	**9**†
	1.9	44	30	36	42	69	610	**9**†
Year	1.1	11,700	698	**232**	3,039	393	1,690	315
	1.5	1,914	410	**176**	549	342	1,564	241
	1.9	266	198	239	279	333	1,283	**193**
Protein	1.1	17,762	5,332	**752**	4,074	1,197	6,572	2,969
	1.5	3,742	1,552	**717**	1,260	1,179	1,822	733
	1.9	692	576	763	738	1,017	1,475	**67**
KEGG	1.1	8,514	**1,406**	1,989	–	4,017	41,801	3,789
	1.5	2,498	848	839	936	1,257	2,473	**681**
	1.9	478	386	451	510	843	1,271	**55**†
CT-Slices	1.1	17,988	4,990	**1,368**	10,677	2,478	20,846	3,635
	1.5	3,202	1,218	**813**	1,524	1,254	1,620	1,159
	1.9	594	484	675	666	978	969	**135**†
Buzz	1.1	–	–	**29,412**	–	–	–	–
	1.5	42,554	40,058	19,332	30,012	39,084	–	**3,923**
	1.9	14,370	14,110	17,887	18,183	24,207	52,942	**1,085**
Attained / 18		17	17	18	16	17	16	17

Bold entries indicate the smallest count among all seven methods in each row, including ties. A dash denotes no attainment within the budget of $60{,}000$ matrix–vector products. Counts include initialization (LSQR for HS-APG-OLS) and all line-search products. Products used solely for objective reporting are excluded. Relative gaps use the common denominator $F(\mathbf{0}) - F_{\mathrm{ref}}$. GAGD+ and UFGM are configured for 10^{-4}. † marks attainment during LSQR initialization.

Table 7: First-hit matrix–vector products for the 18 real-data problems at relative-gap target 10^{-6}.

Dataset	p	HS-APG-F	HS-APG-V	HS-APG-V-BT	GAGD+	UFGM	AGDA	HS-APG-OLS
Power	1.1	5,900	**202**	363	1,509	234	2,241	239
	1.5	724	**124**	148	264	213	2,164	143
	1.9	112	56	71	78	132	918	**27**
Year	1.1	47,044	3,814	**914**	9,279	1,056	6,669	2,441
	1.5	7,290	1,556	536	1,734	780	1,963	**511**
	1.9	780	538	582	633	774	1,646	**269**
Protein	1.1	52,006	16,416	**2,338**	27,231	2,538	16,909	5,209
	1.5	7,622	3,836	**824**	2,490	1,341	2,183	1,353
	1.9	1,430	666	875	1,524	1,167	2,516	**497**
KEGG	1.1	36,628	**9,418**	15,896	–	–	–	42,499
	1.5	8,858	2,896	**2,193**	2,457	3,438	3,811	2,705
	1.9	1,070	896	1,172	1,158	1,935	1,973	**673**
CT-Slices	1.1	–	46,724	**9,878**	–	50,169	–	35,971
	1.5	11,452	6,336	**2,983**	7,662	4,647	23,257	3,589
	1.9	2,484	2,188	2,883	2,805	4,194	6,281	**1,151**
Buzz	1.1	–	–	–	–	–	–	–
	1.5	–	–	–	–	–	–	–
	1.9	58,312	–	–	–	–	–	**11,117**
Attained / 18		15	15	15	13	14	13	16

Bold entries indicate the smallest count among all seven methods in each row, including ties. A dash denotes no attainment within the budget of $60{,}000$ matrix–vector products. Counts include initialization (LSQR for HS-APG-OLS) and all line-search products. Products used solely for objective reporting are excluded. Relative gaps use the common denominator $F(\mathbf{0}) - F_{\mathrm{ref}}$. GAGD+ and UFGM are configured for 10^{-6}.

Convergence trajectories. Figures 3–5 show convergence on all 18 problems, using every recorded output. For the smoothing methods, the curves report the current accelerated output $\boldsymbol{y}_t$; the least-squares-initialized variant also includes the preceding LSQR iterates. AGDA reports the iterate with the lowest objective value so far, as prescribed by the method. No additional running minimum is applied to any curve. The smoothing trajectories are nonmonotone, so reaching a target once does not imply that subsequent outputs remain below it.

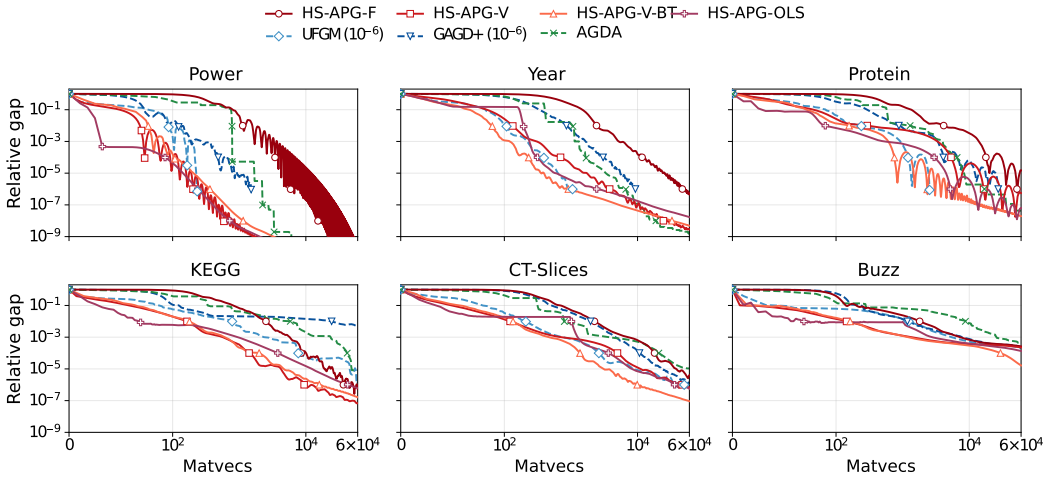

Figure 3: Real-data convergence at $p = 1.1$. Curves show the recorded relative gaps for all seven methods, including LSQR initialization for HS-APG-OLS; UFGM and GAGD+ use target 10^{-6}. All panels share axis scales. The matvec axis is linear from 0 to 10 and logarithmic thereafter. Initialization and line-search products are included; products used only for objective reporting are excluded.

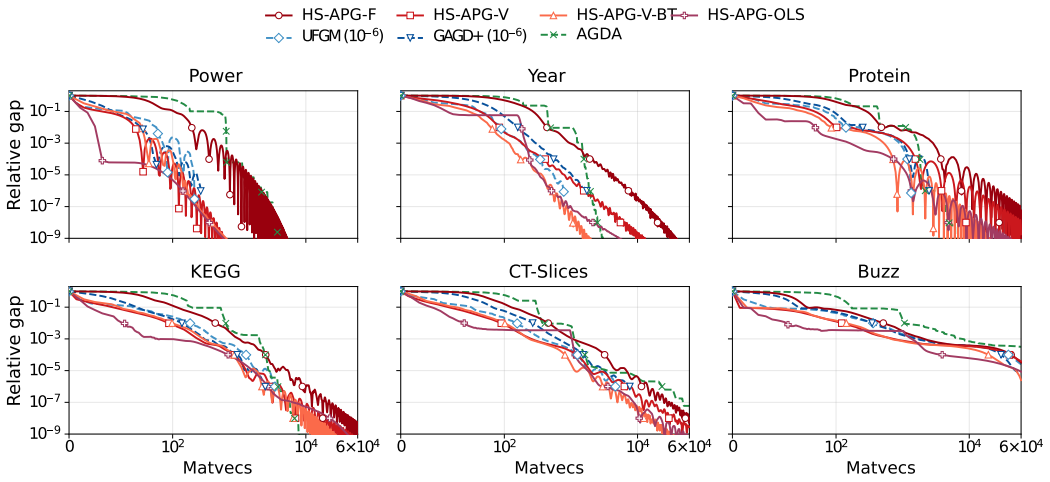

Figure 4: Real-data convergence at $p = 1.5$. Curves show the recorded relative gaps for all seven methods, including LSQR initialization for HS-APG-OLS; UFGM and GAGD+ use target 10^{-6}. All panels share axis scales. The matvec axis is linear from 0 to 10 and logarithmic thereafter. Initialization and line-search products are included; products used only for objective reporting are excluded.

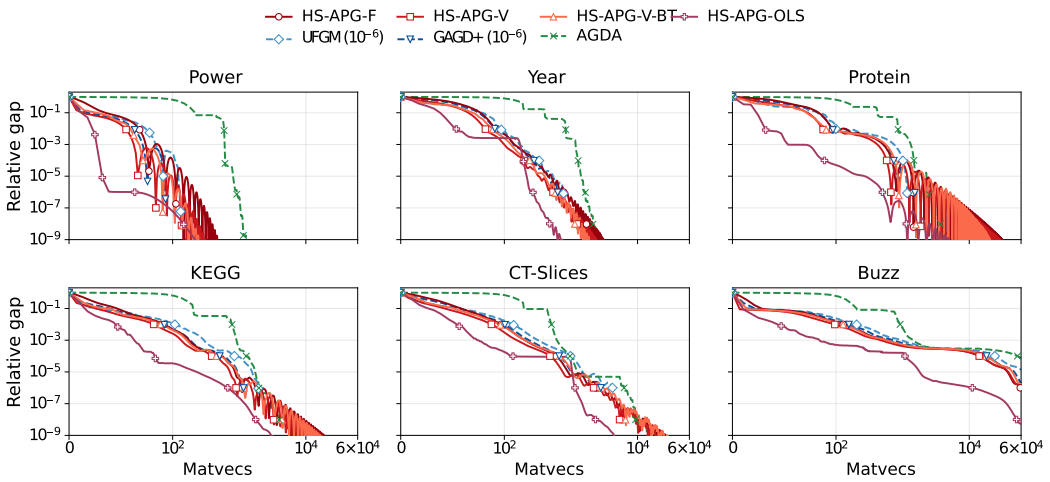

Figure 5: Real-data convergence at $p = 1.9$. Curves show the recorded relative gaps for all seven methods, including LSQR initialization for HS-APG-OLS; UFGM and GAGD+ use target 10^{-6}. All panels share axis scales. The matvec axis is linear from 0 to 10 and logarithmic thereafter. Initialization and line-search products are included; products used only for objective reporting are excluded.

Gradient-restart ablation. The convergence curves in Figures 3–5 exhibit repeated cycles of decrease and rebound. These oscillations motivate adaptive restart, which resets momentum when it appears to oppose descent. We apply the gradient-based rule of O'Donoghue & Candès (2015) to variable smoothing with and without backtracking.

The rule tests whether $\langle \boldsymbol{g}_t, \boldsymbol{y}_t - \boldsymbol{y}_{t-1} \rangle > 0$, where $\boldsymbol{g}_t = \nabla f_{\rho_t}(\boldsymbol{x}_t)$. A positive inner product means that the latest displacement has a component opposite to the negative gradient. The test uses quantities already available in the iteration and requires no additional matrix–vector products, making it inexpensive to add to either method.

The momentum coefficient uses a local iteration counter τ, with $\alpha = 2/(\tau+1)$, while the smoothing radius follows the global iteration count. After at least five local iterations, a positive test resets $\boldsymbol{z}_t \leftarrow \boldsymbol{y}_t$ and $\tau \leftarrow 0$. The current output is retained, as is the accepted curvature estimate in the backtracking variant. Restart therefore does not reject an uphill step or enforce monotonicity.

We compare each method with and without restart on all 18 real-data problems, holding all other settings fixed, including the budget of 60,000 matrix–vector products. Restart is used only in this ablation. The adaptive rule differs from the horizon-doubling scheme in Appendix B.1 and is not covered by the convergence guarantees established here.

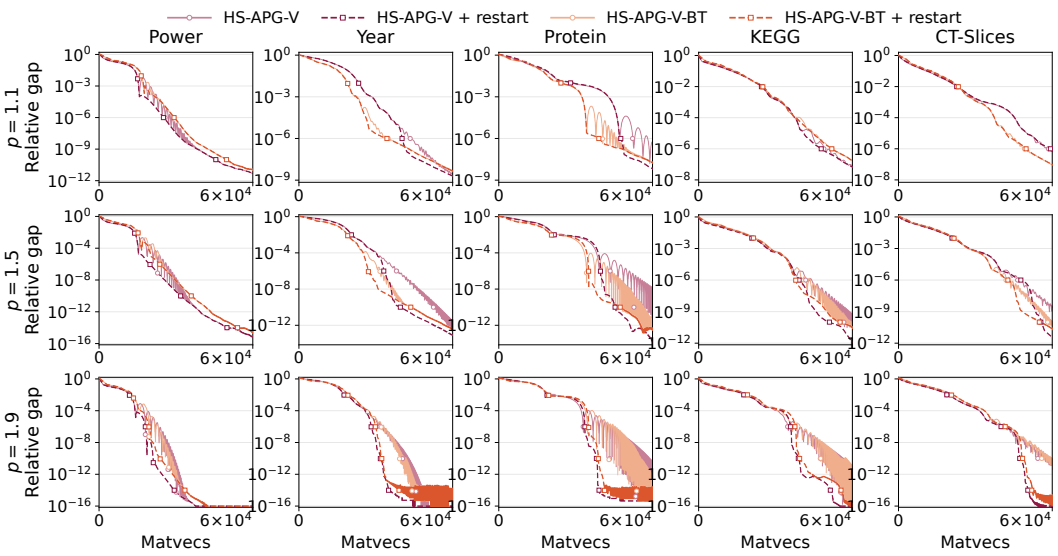

Figure 6: Gradient-restart ablation for HS-APG-V and HS-APG-V-BT. Rows correspond to $p \in \{1.1, 1.5, 1.9\}$; columns show the five datasets on which restart is triggered for at least one configuration. Curves retain all recorded outputs, with gaps below 10^{-16} displayed at that level. Vertical limits vary by panel; the matvec axis is linear from 0 to 10 and logarithmic thereafter.

Figure 6 shows that this simple modification can reduce oscillations and improve terminal accuracy. For variable smoothing on KEGG and CT-Slices at $p = 1.5$, the terminal relative gaps decrease from 4.79×10^{-10} to 1.45×10^{-12} and from 2.88×10^{-9} to 3.22×10^{-12}, respectively. These examples illustrate the value of restart as an inexpensive addition to the smoothing methods when oscillations are pronounced.

Across all 18 problems, restart gives lower, higher, and identical terminal gaps on 10, 4, and 4 problems for variable smoothing, and on 11, 3, and 4 for variable smoothing with backtracking. These counts compare the recorded endpoints directly and include differences near numerical precision, so they do not imply uniform improvement. Buzz remains in these counts but is omitted from the figure: none of its six method–p configurations triggers restart, and each trajectory is identical to its counterpart without restart.

C.3.5 SENSITIVITY TO THE PRODUCT BUDGET AND SMOOTHING HORIZON

The main experiments use a common budget of $B_0 = 60{,}000$ matrix–vector products within the available computational resources. To check whether this cutoff captures the main performance differences, the budget is doubled on four problems selected from the original results. Among the five problems where some methods have not attained 10^{-6}, KEGG at $p = 1.1$ has baselines improving near the cutoff, CT-Slices at $p = 1.1$ has both fixed smoothing and GAGD+ close to the target, and Buzz at $p = 1.9$ has several methods approaching it together. Protein at $p = 1.5$, where all seven methods attain the target within 7,622 products, provides a comparison after early attainment. Selection accounts for additional computational cost and is fixed before the extension; each problem retains all seven methods.

Previously unresolved configurations and all Protein configurations are rerun from their original starting points with a 120,000-product cap. The main runs retain the original parameters, including the fixed-smoothing radius set with $T = 30{,}000$, reference bounds, and product-counting rules. Established first-hit costs on the other three problems are reused.

Table 8: First-hit costs in the budget-sensitivity study, measured in matrix–vector products. Costs without † or ‡ are at most 60,000; † denotes $(60{,}000, 90{,}000]$ and ‡ denotes $(90{,}000, 120{,}000]$. A dash denotes a non-hit by 120,000. Superscript h marks a reused historical event; all other entries come from new cold runs. UFGM and GAGD+ use target-matched configurations. Fixed smoothing retains $T = 30{,}000$. Protein is the solved comparison problem.

Problem	F	V	V-BT	OLS	UFGM	GAGD+	AGDA
Relative-gap target 10^{-4}							
KEGG, $p = 1.1$	8,514^{h}	1,406^{h}	1,989^{h}	3,789^{h}	4,017^{h}	–	41,801
CT-Slices, $p = 1.1$	17,988	4,990^{h}	1,368^{h}	3,635^{h}	2,478^{h}	10,677^{h}	20,846
Buzz, $p = 1.9$	14,370^{h}	14,110	17,887	1,085^{h}	24,207^{h}	18,183^{h}	52,942
Protein, $p = 1.5$	3,742	1,552	717	733	1,179	1,260	1,822
Relative-gap target 10^{-6}							
KEGG, $p = 1.1$	36,628^{h}	9,418^{h}	15,896^{h}	42,499^{h}	76,377†	–	–
CT-Slices, $p = 1.1$	81,864†	46,724^{h}	9,878^{h}	35,971^{h}	50,169^{h}	62,250†	–
Buzz, $p = 1.9$	58,312^{h}	60,412†	70,042†	11,117^{h}	98,910‡	74,817†	–
Protein, $p = 1.5$	7,622	3,836	824	1,353	1,341	2,490	2,183

On the three difficult problems, attainment at 10^{-4} remains 20 of 21 method–problem pairs, while attainment at 10^{-6} increases from 10 to 17 (Table 8). The additional baseline attainments narrow the coverage differences, but a smoothing variant remains fastest at both targets on each selected problem. At 10^{-6}, the lowest smoothing costs are 9,418, 9,878, and 11,117 products on KEGG, CT-Slices, and Buzz, compared with 76,377, 50,169, and 74,817 for the fastest baselines. These cases support the original budget as a practical basis for the target-cost comparisons, while the extension records additional, more costly attainments.

Four additional fixed-smoothing runs set the radius with $T = 60{,}000$, separating a longer run from a change in radius (Table 9). This larger horizon delays both target hits on every selected problem and reverses the ordering of fixed smoothing and OLS on KEGG at 10^{-6}. The comparison concerns first-hit costs; the theoretical radius is chosen for terminal accuracy.

Table 9: Fixed-smoothing first-hit costs under two radius horizons, with a maximum product budget of 120,000. Each horizon has its own trajectory. Both targets are evaluated on that trajectory.

	Target 10^{-4}		Target 10^{-6}	
Problem	$T = 30{,}000$	$T = 60{,}000$	$T = 30{,}000$	$T = 60{,}000$
KEGG, $p = 1.1$	8,514	11,654	36,628	49,960
CT-Slices, $p = 1.1$	17,988	24,594	81,864	111,764
Buzz, $p = 1.9$	14,370	14,876	58,312	60,370
Protein, $p = 1.5$	3,742	4,454	7,622	9,070

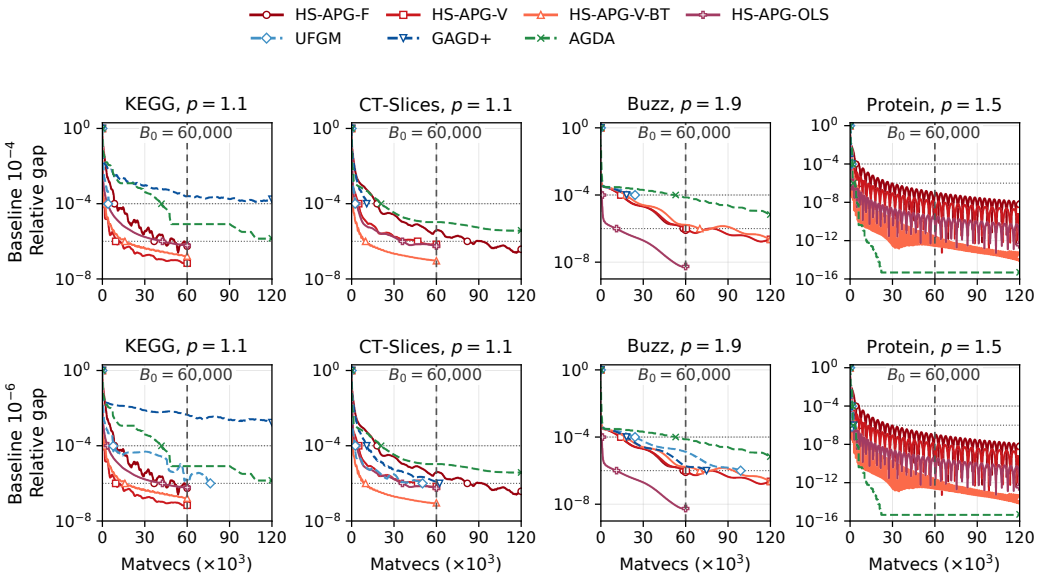

Figure 7: Recorded trajectories under the original parameters. Rows use baseline targets 10^{-4} and 10^{-6}; target-independent methods share their trajectory across rows. Vertical dashed lines mark $B_0 = 60{,}000$, and horizontal dotted lines mark the targets. The four HS-APG curves on KEGG, V/V-BT/OLS on CT-Slices, and F/OLS on Buzz are historical runs ending at the original budget. Previously unresolved configurations and all Protein configurations use a 120,000-product cap. UFGM and GAGD+ stop at the row's target or budget limit. All curves end at their last recorded output; symbols mark initial outputs, first hits, and final outputs. All recorded points are shown, with gaps below 10^{-16} displayed at that level. The x axis is linear in thousands of products; the y axis is logarithmic.

Protein distinguishes target attainment from subsequent accuracy (Figure 7). V-BT reaches 10^{-6} in 824 products, compared with 2,183 for AGDA. AGDA reaches a lower recorded gap of 4.91×10^{-16} by 22,207 products, before the original cutoff, and retains it through the extended run. Its curve follows the prescribed best-iterate rule; the smoothing curves show their current outputs.

C.4 Regularization and constrained experiments

The common algorithm settings and evaluation rules are given in Appendix C.1.

C.4.1 ℓ_1-regularized regression on E2006

Setup. We use the E2006-tfidf and E2006-log1p training sets, each with $16{,}087$ observations. Removing all-zero training columns leaves $150{,}348$ and $4{,}265{,}669$ features, respectively. Each retained feature is divided by its training ℓ_2 norm without centering, while the response is centered and scaled to unit root-mean-square deviation. An intercept is appended, and the augmented design is divided by a conservative spectral-norm upper bound, allowing $\Lambda = 1$ in the algorithmic parameters. All preprocessing statistics are computed from the training set.

We solve (1) with $p = 1.5$ and $\omega(\boldsymbol{u}) = \lambda\|\boldsymbol{w}\|_1$, where $\boldsymbol{u} = (\boldsymbol{w}, c)$, $\boldsymbol{w}$ contains the feature coefficients, and c is the unpenalized intercept. After normalization, the augmented design has the form $\boldsymbol{A} = [\boldsymbol{A}_w, s\mathbf{1}]$, where s is the intercept-column scale and the prediction intercept is sc. Set $c_0 = \arg\min_c f(\mathbf{0}, c)$ and $\boldsymbol{r}_0 = sc_0\mathbf{1} - \boldsymbol{b}$. The smallest penalty for which $(\mathbf{0}, c_0)$ is optimal is

$$\lambda_{\max} = \|\boldsymbol{A}_w^\top \psi'(\boldsymbol{r}_0)\|_\infty, \qquad \psi'(r) = \operatorname{sign}(r)|r|^{1/2}, \tag{92}$$

where ψ' acts coordinatewise. We use $\lambda/\lambda_{\max} \in \{0.01, 0.03, 0.1, 0.3\}$ and initialize every method at $\boldsymbol{u}_0 = (\mathbf{0}, c_0)$. The eight problems and six configurations give 48 runs.

Algorithm settings. Each run has a budget of $30{,}000$ matrix–vector products for TFIDF or $12{,}000$ for log1p. Variable smoothing, with and without backtracking, runs until the budget is exhausted. UFGM and GAGD+ stop at their configured target or budget, whichever comes first. The baseline parameters and common evaluation rules are given in Appendix C.1.

Reference construction. For reference computation, we combine identical feature columns and sum their coefficients. This preserves predictions and cannot increase the ℓ_1 penalty. Conversely, assigning each reduced coefficient to one original column preserves the objective value, so the reduced and original problems have the same optimal value. The compared methods use the original design matrices. We use dual augmented-Lagrangian refinement for lower-penalty problems and high-precision refinement for candidates with small supports. For $p = 1.5$, the Fenchel dual is

$$\max_{\boldsymbol{v}} \; -\boldsymbol{b}^\top \boldsymbol{v} - \frac{1}{3}\|\boldsymbol{v}\|_3^3, \qquad \|\boldsymbol{A}_w^\top \boldsymbol{v}\|_\infty \le \lambda, \quad \mathbf{1}^\top \boldsymbol{v} = 0, \tag{93}$$

We compute primal upper bounds U and dual lower bounds L on the original problems using 256-bit directed rounding. The lower bound includes a correction for the residual in the intercept equality constraint. Every reference satisfies $(U - L)/(F(\boldsymbol{u}_0) - L) \le 10^{-10}$; the largest normalized width is less than 6.56×10^{-13}. We use $F_{\mathrm{ref}} = L$. Table 10 reports the reference values and interval widths.

Distance proxy. As in the unregularized experiments, we use the distance from the initial point to a reference candidate. Since E2006 contains many identical feature columns, we first preprocess the reference candidate by distributing each group's total coefficient equally across its columns. Let $\boldsymbol{u}_{\mathrm{ref}} = (\boldsymbol{w}_{\mathrm{ref}}, c_{\mathrm{ref}})$ be the reference candidate before redistribution. For each group G_g of identical feature columns, define $k_g = |G_g|$ and $z_g = \sum_{j \in G_g} (w_{\mathrm{ref}})_j$. The resulting distance from $\boldsymbol{u}_0$ is

$$D_{\mathrm{proxy}} = \left(\sum_g \frac{z_g^2}{k_g} + (c_{\mathrm{ref}} - c_0)^2 \right)^{1/2}. \tag{94}$$

We use this distance as an empirical estimate, setting $R = D_{\mathrm{proxy}}$ for variable smoothing with and without backtracking.

Table 10: E2006 reference lower bounds F_{ref}, normalized interval widths, and distance proxies D_{proxy}. All eight normalized widths are below 10^{-10}.

Representation	$\lambda/\lambda_{\max}$	F_{ref}	Certified width	D_{proxy}
TFIDF	0.01	1683.142621	4.7547×10^{-13}	39293.7320
TFIDF	0.03	3477.862255	6.5554×10^{-13}	50526.3996
TFIDF	0.1	5096.030672	2.2936×10^{-16}	51425.7842
TFIDF	0.3	7567.517556	6.0878×10^{-16}	24027.6634
log1p	0.01	1334.663936	4.8745×10^{-13}	47102.5470
log1p	0.03	3239.009632	4.2870×10^{-13}	54585.3624
log1p	0.1	5159.912785	2.3311×10^{-16}	54754.1167
log1p	0.3	7525.174126	1.1840×10^{-15}	10829.1211

Results. Table 2 reports the terminal relative gaps. All 48 runs exhaust their matrix–vector budgets; none reaches relative gap 10^{-4} or 10^{-6}. Variable smoothing attains the smallest terminal gap on five problems: TFIDF at ratios 0.01, 0.03, and 0.1, and log1p at 0.01 and 0.03. Variable smoothing with backtracking attains the smallest gap on the remaining three. Across all configurations, terminal relative gaps range from 4.6715×10^{-4} to 8.4810×10^{-2}.

We measure the sparsity of the terminal feature coefficients by

$$s_\eta(\boldsymbol{w}) = \#\left\{j : |w_j| > \eta \max\{1, \|\boldsymbol{w}\|_\infty\}\right\}. \tag{95}$$

Table 11 reports $s_{10^{-8}}$, with $s_{10^{-6}}$ as a sensitivity check. For variable smoothing, the TFIDF counts decrease from $62{,}550$ to $16{,}668$, 830, and 267 as the penalty ratio increases. The corresponding log1p counts are $343{,}331$, $69{,}985$, $2{,}255$, and $1{,}635$.

Table 11: Thresholded active feature counts at termination for all 48 runs. Columns give $\lambda/\lambda_{\max}$. Entries report $s_{10^{-8}}$, with $s_{10^{-6}}$ in parentheses, where $s_\eta = \#\{j : |w_j| > \eta \max(1, \|\boldsymbol{w}\|_\infty)\}$.

	TFIDF			
Method / target	0.01	0.03	0.1	0.3
HS-APG-V	62,550 (32,069)	16,668 (9,149)	830 (349)	267 (105)
HS-APG-V-BT	67,873 (34,953)	16,945 (9,779)	823 (382)	242 (123)
UFGM (10^{-4})	135,355 (72,783)	44,494 (12,931)	2,834 (625)	609 (192)
UFGM (10^{-6})	135,324 (72,702)	44,459 (12,926)	2,825 (624)	609 (192)
GAGD+ (10^{-4})	118,513 (47,599)	34,533 (9,924)	2,209 (424)	544 (146)
GAGD+ (10^{-6})	118,513 (47,599)	34,533 (9,924)	2,209 (424)	544 (146)
	log1p			
Method / target	0.01	0.03	0.1	0.3
HS-APG-V	343,331 (97,795)	69,985 (23,207)	2,255 (726)	1,635 (606)
HS-APG-V-BT	329,372 (109,121)	62,710 (24,906)	1,870 (711)	1,212 (510)
UFGM (10^{-4})	3,105,692 (1,020,374)	564,126 (97,112)	14,432 (2,241)	3,840 (934)
UFGM (10^{-6})	3,107,489 (1,021,937)	562,944 (96,912)	14,471 (2,246)	3,838 (934)
GAGD+ (10^{-4})	2,310,619 (527,518)	325,926 (48,510)	8,217 (1,210)	3,161 (621)
GAGD+ (10^{-6})	2,312,722 (528,050)	325,926 (48,510)	8,217 (1,210)	3,161 (621)

C.4.2 Box-constrained image deblurring

Setup. We use the 512×512 *camera* image from scikit-image v0.25.2 (Van der Walt et al., 2014). Averaging nonoverlapping 4×4 blocks and scaling the intensities to $[0, 1]$ gives a 128×128 image $\boldsymbol{u}_{\mathrm{true}}$. The observation is $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{u}_{\mathrm{true}} + \boldsymbol{\xi}$, where $\boldsymbol{A}$ is periodic convolution with a normalized 5×5 Gaussian kernel of standard deviation 0.6 pixels. The noise entries are independent Gaussian variables with standard deviation 0.01, generated with seed 20260919.

We solve (1) with $p = 1.5$, $m = n = 128^2$, and $\omega = \iota_{[0,1]^n}$, whose proximal operator is coordinate-wise projection onto the box. All methods start from $\boldsymbol{u}_0 = \frac{1}{2}\mathbf{1}$. Fixed smoothing uses the geometric

initial-distance bound $D = \sqrt{n}/2 = 64$, while variable smoothing with and without backtracking use the box diameter $R = \sqrt{n} = 128$. The spectral-norm bound is $\Lambda = 1$. Each run has a budget of 20,000 matrix–vector products, and fixed smoothing uses the nominal horizon $T = 10{,}000$. The smoothing methods stop at relative gap 10^{-6}, while UFGM and GAGD+ stop at their respective configured targets.

Reference construction. Projected Newton continuation produces a feasible reference candidate. For any $\boldsymbol{v} \in \mathbb{R}^m$, the box-constrained problem admits the lower bound

$$L(\boldsymbol{v}) = -\frac{1}{3}\|\boldsymbol{v}\|_3^3 - \boldsymbol{b}^\top \boldsymbol{v} + \sum_{j=1}^{n} \min\{0, (\boldsymbol{A}^\top \boldsymbol{v})_j\}. \tag{96}$$

The final term is the minimum of $\langle \boldsymbol{A}^\top \boldsymbol{v}, \boldsymbol{u} \rangle$ over $[0,1]^n$. Evaluating the candidate objective and the dual bound using 256-bit directed rounding gives an interval containing the optimal objective value. Its normalized width is 2.11×10^{-11}, below the common 10^{-10} threshold. We use the lower endpoint as F_{ref}.

Results. Figure 8 shows the convergence curves, and Table 12 reports the cost of reaching each target. Variable smoothing uses the fewest matrix–vector products at both targets, reaching 10^{-4} and 10^{-6} in 44 and 688 products, respectively, compared with 174 and 3,429 for UFGM and 106 and 1,846 for GAGD+. All seven configurations reach their respective stopping targets. Figure 9 displays the original image, simulated observation, reference candidate, and terminal reconstructions.

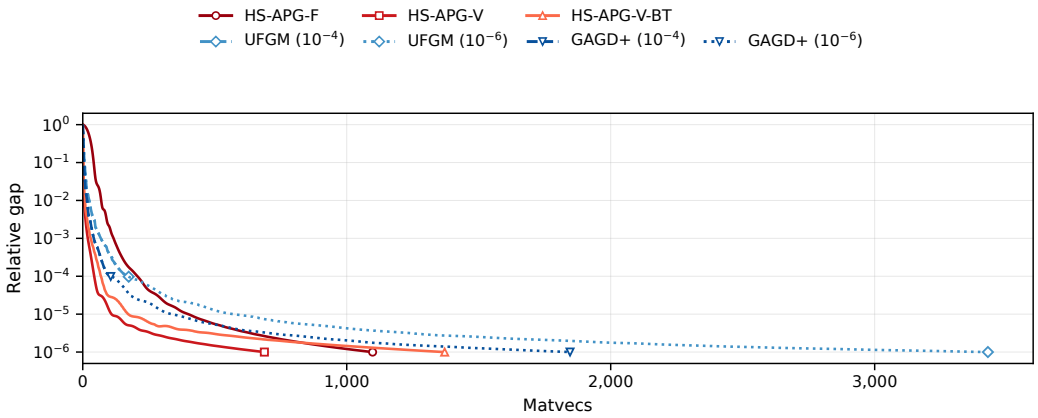

Figure 8: Convergence for box-constrained image deblurring. Curves show relative gaps against matrix–vector products, including line-search trials. Each curve ends at its stopping target. UFGM and GAGD+ configurations are labeled by target.

Table 12: Matrix–vector products required to reach each target in box-constrained image deblurring. UFGM and GAGD+ use the configuration matching each target. Bold entries indicate the smallest count in each column.

Method	Target 10^{-4}	Target 10^{-6}
HS-APG-F	208	1,098
HS-APG-V	**44**	**688**
HS-APG-V-BT	72	1,371
UFGM	174	3,429
GAGD+	106	1,846

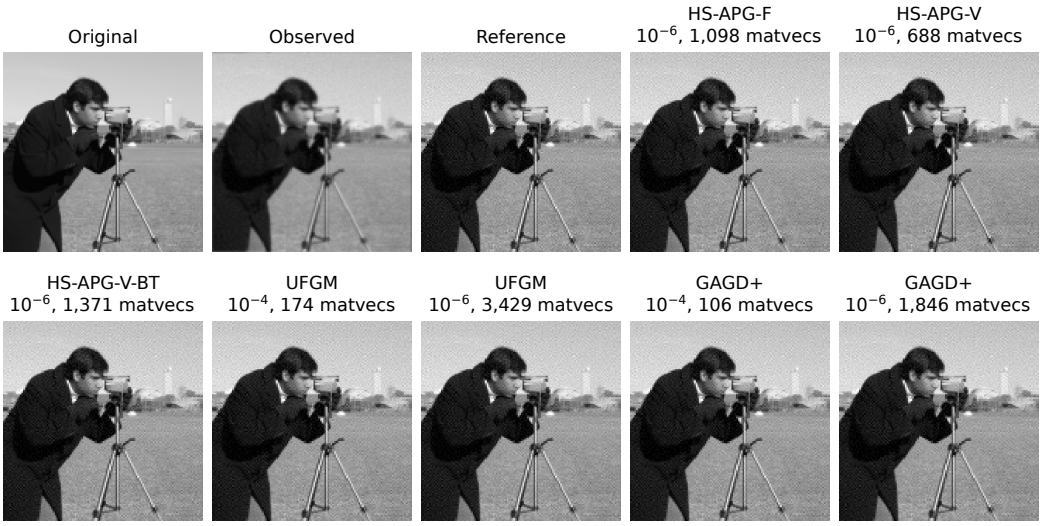

Figure 9: Original image, simulated observation, reference candidate, and terminal reconstructions for box-constrained image deblurring. Method labels give the stopping target and matrix–vector product count. All panels use the grayscale range $[0, 1]$.