History-Aware Adaptive High-Order Tensor Regularization

Chang He * Bo Jiang † Yuntian Jiang ‡ Chuwen Zhang § Shuzhong Zhang ¶

November 17, 2025

Abstract

In this paper, we develop a new adaptive regularization method for minimizing a composite function, which is the sum of a pth-order ($p \geq 1$) Lipschitz continuous function and a simple, convex, and possibly nonsmooth function. We use a history of local Lipschitz estimates to adaptively select the current regularization parameter, an approach we shall term the *history-aware adaptive regularization method*. We explore how the selection of an appropriate volume of historical information affects both the theoretical and practical performance. By using all the historical information, our method matches the complexity guarantees of the standard pth-order tensor methods that require a known Lipschitz constant, for both convex and nonconvex objectives. In the nonconvex case, the number of iterations required to find an (ϵ_g, ϵ_H)-approximate second-order stationary point is bounded by $\mathcal{O}(\max\{\epsilon_g^{-(p+1)/p}, \epsilon_H^{-(p+1)/(p-1)}\})$. For convex functions, we establish an $\mathcal{O}(\epsilon^{-1/p})$ iteration complexity for finding an ϵ-approximate optimal point and further propose an accelerated variant attaining an iteration complexity of $\mathcal{O}(\epsilon^{-1/(p+1)})$. For practical consideration, we propose several variants of this method with only part of historical information. We introduce cyclic and sliding-window strategies for choosing historical Lipschitz estimates, which mitigate the limitation of overly conservative updates. As long as a rough upper bound of the Lipschitz constant is known, these two variants achieve the same iteration complexity guarantees in terms of the input accuracy as the method using full historical information. Finally, extensive numerical experiments are conducted to demonstrate the effectiveness of our adaptive approach.

1 Introduction

1.1 Motivation

Adaptively selecting parameters in optimization methods, such as the stepsize in first-order methods, is a central challenge in both theoretical analysis and practical implementation. Typically, there are three common strategies: line-search procedures, ratio test in trust-region methods, and adaptive regularization. The line-search procedure, which can be dated back to Armijo (1966), was designed for gradient methods. It introduces an inner loop to find a suitable stepsize at each iteration.

*The authors are listed alphabetically. School of Information Management and Engineering, Shanghai University of Finance and Economics; Department of Industrial and System Engineering, University of Minnesota. ischanghe@gmail.com

†School of Information Management and Engineering, Shanghai University of Finance and Economics. isyebojiang@gmail.com

‡School of Information Management and Engineering, Shanghai University of Finance and Economics. yuntianjiang07@gmail.com

§Booth School of Business, University of Chicago. chuwen.zhang@chicagobooth.edu

¶Department of Industrial and System Engineering, University of Minnesota. zhangs@umn.edu

Beyond first-order methods, trust-region methods (Conn et al., 2000) operate on a local quadratic approximation of the objective function within a ball constraint. By introducing a ratio test, the trust-region method adaptively adjusts the radius to guarantee global convergence. These two strategies are mostly designed for specific derivative information, namely, first-order and second-order derivative information. The line-search procedure relies on an explicit update with a closed form, which is generally available in first-order methods, while the trust-region method is based on a quadratic approximation. Unlike these two strategies, the third one that works for updates using arbitrary-order derivative information is called *adaptive regularization*, which is the focus of this work.

The regularization technique was initially proposed for Newton's method (Griewank, 1981) and is well developed in Conn et al. (2000). Nesterov and Polyak (2006) theoretically established the first global complexity guarantees for the cubic regularization of Newton's method, and Weiser et al. (2007) then designed an affine-invariant version with encouraging numerical performance. Subsequently, this regularization was extended to a more general pth-order tensor update (Birgin et al., 2017; Nesterov, 2021):

$$\begin{aligned} d^k &= \operatorname{argmin}_{d\in\mathbb{R}^n} f(x^k) + \sum_{\ell=1}^{p} \frac{1}{\ell!} \mathrm{D}^\ell f(x^k)[d]^{\otimes \ell} + \frac{\sigma_k}{(p+1)!}\|d\|^{p+1}, \\ x^{k+1} &= x^k + d^k, \ \ k = 0, 1, 2, \ldots \end{aligned} \tag{1}$$

where the search direction is generated by minimizing a regularized pth-order Taylor polynomial. To ensure the global convergence of the above scheme, one must carefully select the regularization parameter σ_k at each iteration k, which should be close to the true pth-order Lipschitz constant. Therefore, the technique of adaptive regularization was developed to estimate the unknown Lipschitz constant automatically. Generally, there exists two main approaches. Nesterov and Polyak (2006) adopted the idea of the line-search procedure for first-order methods and proposed a line-search adaptive regularization[1] (`LS-AR`) strategy for the cubic regularization case ($p = 2$). At each iteration, `LS-AR` uses an inner loop to repeatedly increase the value of σ_k by a factor of 2 until a sufficient decrease condition is met. This strategy was later extended to general pth-order tensor updates and to functions with Hölder continuous derivatives. Another approach is the adaptive regularization with cubics (`ARC`) developed by (Grapiglia and Nesterov, 2017, 2020, 2022). Instead of an additional inner loop, `ARC` evaluates the search direction at each iteration using a ratio of the actual decrease in the objective function to the decrease predicted by the regularized Taylor polynomial. The regularization parameter σ_k and iterate x^k are then updated both based on this ratio. This framework was later generalized to adaptive regularization with pth-order models (`ARP`) (Birgin et al., 2017; Cartis et al., 2018, 2022).

These two adaptive regularization schemes are "Markovian", as their parameter selection depends only on information at the current iterate. `LS-AR` introduces an inner loop at the current iterate until a proper estimate is found, and the ratio test of `ARP` is also defined in terms of the current iterate. However, properly using the *history information* can also be beneficial. One evident example is Nesterov's accelerated gradient method (Nesterov, 1983), which uses the previous iterate to gain acceleration. Besides, a recent trend in line-search-free first-order methods[2] (Malitsky and Mishchenko, 2020; Li and Lan, 2025) also demonstrates that the past information can effectively estimate the current parameter. These works naturally motivate us to explore the role of history

[1] We clarify that this is distinct from the line-search procedure commonly used in first-order methods. We adopt the name "line-search adaptive regularization" because the authors of the original work refer to their adaptive parameter selection scheme as a "line-search strategy". (see Section 5.2 of Nesterov and Polyak (2006))

[2] A detailed review is presented in Section 1.3.

information in designing optimization methods. Therefore, in this paper, we propose an approach termed the *History-aware Adaptive Regularization* (HAR) method. This method leverages Lipschitz constant estimates from past iterates to inform the choice of the current regularization parameter. Our results reveal that while using all previous information is theoretically sound, using only part of the historical information by introducing a budget is often more practical. Our main contributions are presented below.

1.2 Main contributions

In this work, motivated by the auto-conditioned gradient descent method (Lan et al., 2024; Li and Lan, 2025; Yagishita and Ito, 2025), we develop the history-aware adaptive regularization method (Algorithm 1) in parallel with LS-AR and ARP. It is a unified adaptive regularization scheme that works for both convex and nonconvex objectives. Theoretically, we establish the iteration complexity of HAR for the composite objective in problem (2). When the differentiable part is convex, HAR finds an ϵ-approximate optimal point (see Definition 2.1) in $\mathcal{O}(\epsilon^{-1/p})$ iterations. Furthermore, an accelerated version, HAR-A (Algorithm 4), is also proposed for this case, achieving an improved $\mathcal{O}(\epsilon^{-1/(p+1)})$ iteration complexity. For a nonconvex differentiable part, HAR finds an (ϵ_g, ϵ_H)-approximate second-order stationary point (see Definition 2.2) in at most $\mathcal{O}(\max\{\epsilon_g^{-(p+1)/p}, \epsilon_H^{-(p+1)/(p-1)}\})$ iterations. These results match those of standard pth-order tensor methods when the true Lipschitz constant is given.

On the practical side, we propose two variants: HAR-C (Algorithm 2) and HAR-S (Algorithm 3). The original HAR utilizes the local estimates from all the past iterates, which is sometimes not efficient in practice. Therefore, these two variants introduce a predefined budget to limit the number of historical local estimates. This strategy remedies potential overly conservative updates in the practical application of HAR. Under the condition that a rough upper bound of the pth-order Lipschitz constant is known, we establish convergence guarantees for these variants that match the original HAR in terms of the input accuracy. To validate the adaptivity of our proposed methods, we test the two practical variants, HAR-C and HAR-S, on numerous applications, ranging from convex to nonconvex problems. With a properly selected budget, our numerical performance shows that both methods, especially HAR-S, are highly comparable to the state-of-the-art implementation of ARC (Dussault et al., 2024).

1.3 Related works

In this subsection, we briefly review existing works related to our approach. The first part covers general pth-order updates (or high-order methods), which is the setting considered in this work. The second part discusses recently popular line-search-free first-order methods, which motivated us to explore how the selection of an appropriate volume of historical information affects both the theoretical and practical performance of our approach.

High-order optimization methods. Beyond popular first-order methods (Beck, 2017), high-order optimization methods have drawn significant attention over the decades. The seminal work of Nesterov and Polyak (2006) provided the first global convergence rate for a second-order method, which was followed by the development of the more practical version ARC (Cartis et al., 2011a,b). These pioneering works motivated a trend in the complexity analysis of various second-order methods, including trust-region type methods (Curtis et al., 2017, 2021; Jiang et al., 2023, 2025), homogeneous second-order methods (Zhang et al., 2025; He et al., 2025b), gradient regularized Newton's methods (Mishchenko, 2023; Doikov and Nesterov, 2024; Doikov et al., 2024), and Newton-CG methods

(Royer et al., 2020; Yao et al., 2023; He et al., 2025a). Extending the analysis to derivatives of order higher than two, Birgin et al. (2017); Cartis et al. (2018) proposed ARP and analyzed its iteration complexity for nonconvex optimization. However, a major challenge with this high-order approach was that it was unknown how to efficiently solve the subproblem. A breakthrough for the convex case came when Nesterov (2021) proposed an implementable third-order tensor method. More recently, third-order subproblem solvers for nonconvex objectives have also been developed (Cartis et al., 2024; Zhu and Cartis, 2025; Cartis and Zhu, 2025b,a; Zhou et al., 2025b). In addition, numerous variants have been proposed for achieving optimal complexity in convex optimization (Monteiro and Svaiter, 2013; Gasnikov et al., 2019; Kovalev and Gasnikov, 2022; Carmon et al., 2022), for practical considerations (Jiang et al., 2020; Huang et al., 2024; Xu et al., 2020), or as stochastic extensions (Hanzely et al., 2020; Chen et al., 2022; Lucchi and Kohler, 2023).

Line-search-free first-order methods. Line-search-free first-order methods have become an active area of research in recent years. These methods employ adaptive stepsize rules that estimate the Lipschitz constant using information from history iterates, avoiding the inner loops required by traditional line-search procedures. This trend was initiated by two main strategies. The first was proposed by Malitsky and Mishchenko (2020) for convex objectives. The second is the auto-conditioned stepsize, introduced by Li and Lan (2025) with an accelerated version for convex optimization (which also works for Hölder smoothness). Both strategies have motivated numerous variants. The Malitsky-Mishchenko stepsize was extended to the proximal setting for convex problems (Malitsky and Mishchenko, 2024; Latafat et al., 2024; Oikonomidis et al., 2024) and nonconvex proximal objectives (Ye et al., 2025), as well as to stochastic optimization (Aujol et al., 2025). Similarly, the auto-conditioned stepsize was also shown to work in the proximal and stochastic settings (Yagishita and Ito, 2025; Lan et al., 2024). Beyond these two main approaches, other adaptive strategies have also been developed. For example, Zhou et al. (2025a) proposed a strategy based on the traditional BB stepsize (Barzilai and Borwein, 1988); Gao et al. (2024) developed hypergradient stepsize choice inspired by online learning; Suh and Ma (2025) designed an adaptive accelerated gradient method for convex objectives with convergence guarantees on both the function value gap and the gradient norm.

2 Preliminaries

In this paper, we study the following composite optimization problem:

$$\min_{x\in\mathrm{dom}(\psi)} \quad F(x) = f(x) + \psi(x), \tag{2}$$

where function $f(\cdot)$ is pth-order continuously differentiable, and $\psi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is a simple proper closed convex function with $\mathrm{dom}(\psi) \subseteq \mathbb{R}^n$. We assume that this problem admits at least one solution, which we denote by $x^\star \in \mathrm{dom}(\psi)$. Since ψ and F are possibly nonsmooth, we use $\partial\psi(x)$ and $\partial F(x)$ to denote their respective subgradients at a point x. Given a point $x \in \mathbb{R}^n$ and a nonempty subset $\mathcal{S} \subseteq \mathbb{R}^n$, we denote by $\mathrm{dist}(x, \mathcal{S}) \coloneqq \inf_{y\in\mathcal{S}} \|x - y\|$ the distance between them. Given a symmetric matrix $A \in \mathbb{R}^{n\times n}$, we use $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ to denote its smallest and largest eigenvalues, respectively. The identity matrix is denoted by $\mathrm{I}_{n\times n}$. Now we introduce several approximate optimality measures for problem (2).

Definition 2.1. *Given an accuracy $\epsilon > 0$, we say point x is an ϵ-approximate optimal point if it satisfies $F(x) - F(x^\star) \le \epsilon$.*

Definition 2.2. *Given accuracies $\epsilon_g, \epsilon_H > 0$, we say point x is an ϵ_g-approximate first-order stationary point if it satisfies* $\operatorname{dist}(x, \partial F(x)) \le \epsilon_g$. *Furthermore, when $\psi \equiv 0$, we say point x is an ϵ_H-approximate second-order stationary point if it additionally satisfies* $\lambda_{\min}(\nabla^2 F(x)) \ge -\epsilon_H$.

For $p \ge 1$, the pth-order directional derivative of function f at point x along directions $h_i \in \mathbb{R}^n$, $i = 1, \ldots, p$ is denoted by $\mathrm{D}^p f(x)[h_1, \ldots, h_p]$. If all directions $h_1, \ldots, h_p$ are the same, we apply a simpler notation $\mathrm{D}^p f(x)[h]^{\otimes p}$, $h \in \mathbb{R}^n$. In particular, for any $x \in \mathbb{R}^n$ and $h_1, h_2 \in \mathbb{R}^n$, we have

$$\mathrm{D} f(x)[h_1] = \langle \nabla f(x), h_1 \rangle, \ \mathrm{D}^2 f(x)[h_1, h_2] = \langle \nabla^2 f(x) h_1, h_2 \rangle.$$

Then the operator norm of $\mathrm{D}^p f(x)$ is defined by

$$\| \mathrm{D}^p f(x) \| = \max_{\|h\| \le 1} \left| \mathrm{D}^p f(x)[h]^{\otimes p} \right|.$$

The pth-order Taylor polynomial of function f at point x is defined as follows:

$$\mathrm{T}_p(y; x) = f(x) + \sum_{\ell=1}^{p} \frac{1}{\ell!} \mathrm{D}^\ell f(x)[y - x]^{\otimes \ell}, \ \forall y \in \mathbb{R}^n.$$

We use $\Omega_p^\sigma(y; x)$ to denote the pth-order Taylor polynomial regularized by a $(p+1)$th-order term with parameter σ:

$$\Omega_p^\sigma(y; x) = \mathrm{T}_p(y; x) + \frac{\sigma}{(p+1)!} \|y - x\|^{p+1}, \ \forall y \in \mathbb{R}^n.$$

We say the pth-order direction derivative of function f is L_p-Lipschitz continuous if it satisfies

$$\| \mathrm{D}^p f(x) - \mathrm{D}^p f(y) \| \le L_p \|x - y\|, \ \forall x, y \in \mathbb{R}^n. \tag{3}$$

Under the above Lipschitz continuous property, several Lipschitz-type inequalities hold. For any $p \ge 1$, the function value satisfies

$$|f(y) - \mathrm{T}_p(y; x)| \le \frac{L_p}{(p+1)!} \|y - x\|^{p+1}, \ \forall x, y \in \mathbb{R}^n. \tag{4}$$

If $p \ge 2$, the gradient satisfies

$$\|\nabla f(y) - \nabla \mathrm{T}_p(y; x)\| \le \frac{L_p}{p!} \|y - x\|^p, \tag{5}$$

and the Hessian matrix satisfies

$$\|\nabla^2 f(y) - \nabla^2 \mathrm{T}_p(y; x)\| \le \frac{L_p}{(p-1)!} \|y - x\|^{p-1}. \tag{6}$$

The proof of the above inequalities can be found in Birgin et al. (2017); Cartis et al. (2018). Finally, we call a differentiable function $d(x)$ on $\mathbb{R}^n$ uniformly convex (see Section 4.2.2 in Nesterov (2018)) of degree $p \ge 2$ with constant $q > 0$ if

$$d(y) \ge d(x) + \langle \nabla d(x), y - x \rangle + \frac{q}{p} \|y - x\|^p, \ \forall x, y \in \mathbb{R}^n. \tag{7}$$

3 Algorithm Design and Convergence Analysis

In this section, we first provide the algorithm design of HAR. Similar to standard pth-order tensor methods, HAR finds a search direction d^k by minimizing a regularized pth-order Taylor polynomial at iteration k (Line 5 of Algorithm 1). Then a local Lipschitz constant estimate H_k is calculated, which uses the ratio of the Taylor remainder to the magnitude of the search direction:

$$\frac{H_k}{(p+1)!} = \frac{f(x^{k-1}+d^k) - \mathrm{T}_p(x^{k-1}+d^k; x^{k-1})}{\|d^k\|^{p+1}}.$$

Motivated by the auto-conditioned stepsize strategy, we introduce an adaptive parameter M_k as the maximum of all historical local Lipschitz estimates. The final regularization parameter is set as $\sigma_k = \alpha M_k$, where $\alpha > 1$ is a predefined parameter. Compared with the gradient descent method with auto-conditioned stepsize, we introduce a monotone step in HAR (Line 7 in Algorithm 1). Since the adaptive parameter M_k may not be a close approximation of the true Lipschitz constant at some iterations, the direction generated in this scenario could unexpectedly increase the objective value. The monotone step acts as a safeguard: if a sufficient decrease is not achieved, a null iteration is taken instead. This mechanism prevents any undesirable increases in the function value, which is necessary for the convergence analysis.

Algorithm 1: History-Aware Adaptive Regularization Method

1: **Input:** Initial point $x^0 \in \mathrm{dom}(\psi)$, initial guess $H_0 = M_0 > 0$, parameter $\alpha > 1$
2: **For** $k = 1, 2, \ldots$ **do**
3: Update the adaptive parameter $M_k = \max\{M_{k-1}, H_{k-1}\}$;
4: Calculate the regularization parameter $\sigma_k = \alpha M_k$;
5: Compute the search direction

$$d^k = \mathrm{argmin}_{d \in \mathrm{dom}(\psi)} \, \Omega_p^{\sigma_k}(x^{k-1}+d; x^{k-1}) + \psi(x^{k-1}+d); \tag{8}$$

6: Update the intermediate iterate $x^{k-0.5} = x^{k-1} + d^k$;
7: Update the iterate $x^k = \mathrm{argmin}_{x \in \{x^{k-0.5}, x^{k-1}\}} F(x)$;
8: Update the local Lipschitz estimate

$$\frac{H_k}{(p+1)!} = \frac{f(x^{k-0.5}) - \mathrm{T}_p(x^{k-1}+d^k; x^{k-1})}{\|d^k\|^{p+1}};$$

9: **End For**

By the local Lipschitz estimate in Line 8, we see

$$f(x^{k-0.5}) - \mathrm{T}_p(x^{k-1}+d^k; x^{k-1}) = \frac{H_k}{(p+1)!}\|d^k\|^{p+1}. \tag{9}$$

The optimality condition of the subproblem (8) follows

$$\Omega_p^{\sigma_k}(x^{k-1}+d^k; x^{k-1}) + \psi(x^{k-1}+d^k) \le \Omega_p^{\sigma_k}(x^{k-1}; x^{k-1}) + \psi(x^{k-1}) = f(x^{k-1}) + \psi(x^{k-1}). \tag{10}$$

By combining the above two equations (10), the function value satisfies

$$f(x^{k-0.5}) \le f(x^{k-1}) + \psi(x^{k-1}) - \frac{\sigma_k - H_k}{(p+1)!}\|d^k\|^{p+1} - \psi(x^{k-1}+d^k),$$

which can be written equivalently as

$$F(x^{k-0.5}) - F(x^{k-1}) \leq -\frac{\sigma_k - H_k}{(p+1)!}\|d^k\|^{p+1}. \tag{11}$$

The above descent inequality naturally motivates dividing the sequence of iterates into two categories: successful and unsuccessful. We define these two index sets mathematically as follows

$$\mathcal{S} := \{k \geq 1 \mid (\alpha+1)M_k \geq 2H_k\}, \text{ and } \mathcal{U} := \mathbb{N}\backslash\mathcal{S}. \tag{12}$$

Clearly, for any $k \geq 1$, it holds that $k = |\mathcal{S} \cap \{1, \ldots, k\}| + |\mathcal{U} \cap \{1, \ldots, k\}|$. To refer to the elements of these sets, we let $\mathcal{S}(j)$ denote the index of the jth successful iteration (and similarly for $\mathcal{U}(j)$). An iteration is successful because the function value is guaranteed to decrease sufficiently:

$$F(x^{k-0.5}) - F(x^{k-1}) \leq -\frac{(\alpha-1)M_k}{2(p+1)!}\|d^k\|^{p+1}, \ \forall k \in \mathcal{S}.$$

Conversely, for an unsuccessful iteration $k \in \mathcal{U}$, a sufficient decrease is not guaranteed, though a decrease may still occur. All null iterations induced by the monotone step are therefore contained within the set of unsuccessful iterations $\mathcal{U}$. A technical result of our analysis is that the number of unsuccessful iterations is finite, which is motivated by Yagishita and Ito (2025). The intuition is that `HAR` performs an implicit tuning during each unsuccessful iteration: the local estimate is implicitly increased, which eventually forces a successful one. This finiteness and the monotonicity together guarantee the convergence of `HAR`. The following lemma also shows that the choice of the parameter α involves a trade-off between reducing the number of unsuccessful iterations and avoiding overly conservative update in practice.

Lemma 3.1. *Suppose that function f satisfies (3). The cardinality of the unsuccessful index set $\mathcal{U}$ in Algorithm 1 satisfies*

$$|\mathcal{U}| \leq \overline{\mathcal{U}} := \left\lceil \log_{\frac{\alpha+1}{2}} \frac{H_{\max}}{H_0} \right\rceil,$$

where $H_{\max} = \max\{H_0, L_p\}$.

Proof. Note that for any unsuccessful iteration $\mathcal{U}(j)$, we have $(\alpha+1)M_{\mathcal{U}(j)} < 2H_{\mathcal{U}(j)}$, $j \geq 1$. Recall that $M_{\mathcal{U}(j)+1} = \max\{H_0, H_1, \ldots, H_{\mathcal{U}(j)}\}$. It follows

$$M_{\mathcal{U}(j)+1} \geq H_{\mathcal{U}(j)} > \frac{\alpha+1}{2} M_{\mathcal{U}(j)}, \ \forall j \geq 1.$$

Now we prove the result by contradiction. Suppose the cardinality of the unsuccessful index set satisfies $|\mathcal{U}| > \overline{\mathcal{U}}$. On the one hand, for any iteration in the region $J \in [\overline{\mathcal{U}}, |\mathcal{U}|]$, the above inequality implies

$$M_{\mathcal{U}(J)+1} > \left(\frac{\alpha+1}{2}\right)^J M_{\mathcal{U}(1)} \geq \frac{H_{\max}}{H_0} M_{\mathcal{U}(1)}.$$

Since $\mathcal{U}(1) \geq 1$ and the sequence $\{M_k\}$ is nondecreasing, we have $M_{\mathcal{U}(1)} \geq M_1 = H_0$. Substituting this into the inequality above yields $M_{\mathcal{U}(J)+1} > H_{\max}$. On the other hand, from the definition of H_k and inequality (4), we have the following bound for all $k \geq 1$:

$$\frac{H_k}{(p+1)!} = \frac{f(x^{k-0.5}) - \mathrm{T}_p(x^{k-1} + d^k; x^{k-1})}{\|d^k\|^{p+1}} \leq \frac{L_p}{(p+1)!}.$$

This implies that $H_k \leq L_p$ for all $k \geq 1$. By the definition of M_k, we can then conclude that $M_k \leq \max\{L_p, H_0\} = H_{\max}$, $k \geq 1$. Therefore, it holds that $M_{\mathcal{U}(J)+1} \leq H_{\max}$ and $M_{\mathcal{U}(J)+1} > H_{\max}$, which is a contradiction. The proof is completed. □

Complexity analysis for the convex case. Now we establish the iteration complexity of HAR under the condition that the differentiable part f in problem (2) is convex. Our analysis begins with a global perspective on the intermediate iterate in Line 6 of Algorithm 1. First note that Equation (9) implies

$$\begin{aligned} F(x^{k-0.5}) &= f(x^{k-0.5}) + \psi(x^{k-0.5}) \\ &= \mathrm{T}_p(x^{k-1} + d^k; x^{k-1}) + \psi(x^{k-0.5}) + \frac{H_k}{(p+1)!}\|d^k\|^{p+1} \\ &= \Omega_p^{\sigma_k}(x^{k-1} + d^k; x^{k-1}) + \psi(x^{k-1} + d^k) + \frac{H_k - \sigma_k}{(p+1)!}\|d^k\|^{p+1}. \end{aligned}$$

Since the search direction is generated from the subproblem (8), the above equation follows

$$\begin{aligned} F(x^{k-0.5}) + \frac{\sigma_k - H_k}{(p+1)!}\|d^k\|^{p+1} &= \min_{d\in\mathbb{R}^n}\left\{\Omega_p^{\sigma_k}(x^{k-1} + d; x^{k-1}) + \psi(x^{k-1} + d)\right\} \\ &\le \min_{d\in\mathbb{R}^n}\left\{f(x^{k-1} + d) + \psi(x^{k-1} + d) + \frac{\sigma_k + L_p}{(p+1)!}\|d\|^{p+1}\right\}, \end{aligned}$$

where we use inequality (4) in the second step. Recall that for any successful iteration $\mathcal{S}(j)$, $(\alpha+1)M_{\mathcal{S}(j)} \ge 2H_{\mathcal{S}(j)}$ for any $j \ge 1$. This implies $\alpha M_{\mathcal{S}(j)} - H_{\mathcal{S}(j)} \ge (\alpha-1)M_{\mathcal{S}(j)}/2 > 0$. Consequently, we have the following inequality, which is the main tool for establishing the iteration complexity:

$$F(x^{\mathcal{S}(j)}) = F(x^{\mathcal{S}(j)-0.5}) \le \min_{d\in\mathbb{R}^n}\left\{F(x^{\mathcal{S}(j)-1} + d) + \frac{\sigma_{\mathcal{S}(j)} + L_p}{(p+1)!}\|d\|^{p+1}\right\},\ \forall j \ge 1. \tag{13}$$

Theorem 3.1. *Suppose the function f is convex and satisfies (3). Furthermore, assume that the initial sublevel set $\mathcal{F}(x^0) := \{x \in \mathrm{dom}(\psi) \mid F(x) \le F(x^0)\}$ is bounded, i.e., $R := \sup_{x,y\in\mathcal{F}(x^0)}\|x - y\| < \infty$. Then the sequence $\{x^k\}_{k\ge0}$ generated by Algorithm 1 satisfies*

$$F(x^k) - F(x^\star) \le \frac{2\alpha H_{\max} R^{p+1}}{p!}\left(\frac{p+1}{k-\overline{\mathcal{U}}}\right)^p,\ \forall k \ge \overline{\mathcal{U}}.$$

Proof. First note that $\|x^k - x^\star\| \le R$, $\forall k \ge 0$ since Algorithm 1 is monotone. By applying Equation (13) and convexity of F, the function value at any successful iteration $\mathcal{S}(j)$ satisfies

$$\begin{aligned} F(x^{\mathcal{S}(j)}) &\le \min_{\tau\in[0,1]}\left\{F(x^{\mathcal{S}(j)-1} + \tau(x^\star - x^{\mathcal{S}(j)-1})) + \frac{\tau^{p+1}(\sigma_{\mathcal{S}(j)} + L_p)}{(p+1)!}\|x^\star - x^{\mathcal{S}(j)-1}\|^{p+1}\right\} \\ &\le \min_{\tau\in[0,1]}\left\{F(x^{\mathcal{S}(j)-1}) - \tau\big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big) + \frac{\tau^{p+1}(\sigma_{\mathcal{S}(j)} + L_p)R^{p+1}}{(p+1)!}\right\} \\ &\le \min_{\tau\in[0,1]}\left\{F(x^{\mathcal{S}(j)-1}) - \tau\big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big) + \frac{2\tau^{p+1}\alpha H_{\max} R^{p+1}}{(p+1)!}\right\}. \end{aligned} \tag{14}$$

The last step holds because these parameters satisfy $\sigma_{\mathcal{S}(j)} = \alpha M_{\mathcal{S}(j)} \le \alpha H_{\max}$, $L_p \le H_{\max}$ and $\alpha > 1$. Let

$$\varphi_j(\tau) := F(x^{\mathcal{S}(j)-1}) - \tau\big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big) + \frac{2\tau^{p+1}\alpha H_{\max} R^{p+1}}{(p+1)!}.$$

The minimum of the univariate function $\varphi_j(\cdot)$ is achieved at

$$\tau_j^\star = \left(\frac{p!\big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big)}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}}.$$

Choosing $\tau = 1$ in Equation (14) implies

$$F(x^{\mathcal{S}(j)}) \leq F(x^\star) + \frac{2\alpha H_{\max} R^{p+1}}{(p+1)!}, \ \forall j \geq 1. \tag{15}$$

By combining the above inequality with $F(x^{\mathcal{S}(j)-1}) \leq F(x^{\mathcal{S}(j-1)})$, we obtain

$$\tau_j^\star = \left(\frac{p!\big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big)}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}} \leq \left(\frac{p!\big(F(x^{\mathcal{S}(j-1)}) - F(x^\star)\big)}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}} \leq \left(\frac{1}{p+1} \right)^{\frac{1}{p}} \leq 1, \ \forall j \geq 2.$$

Therefore, we are safe to substitute $\tau = \tau_j^\star$ into inequality (14). For any $j \geq 2$, this yields the following inequality

$$F(x^{\mathcal{S}(j)}) - F(x^\star) \leq F(x^{\mathcal{S}(j)-1}) - F(x^\star) - \frac{p}{p+1} \left(\frac{p!}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}} \big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big)^{\frac{p+1}{p}}.$$

Let $\Delta_j \coloneqq F(x^{\mathcal{S}(j)}) - F(x^\star)$. Now we claim the following recursive inequality holds in terms of Δ_j:

$$\Delta_j \leq \Delta_{j-1} - \frac{p}{p+1} \left(\frac{p!}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}} \Delta_{j-1}^{\frac{p+1}{p}}, \ \forall j \geq 2. \tag{16}$$

This is sufficient to show the following inequality holds for any $j \geq 2$:

$$\begin{aligned} &F(x^{\mathcal{S}(j)-1}) - F(x^\star) - \frac{p}{p+1} \left(\frac{p!}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}} \big(F(x^{\mathcal{S}(j)-1}) - F(x^\star)\big)^{\frac{p+1}{p}} \\ \leq\ &\Delta_{j-1} - \underbrace{\frac{p}{p+1} \left(\frac{p!}{2\alpha H_{\max} R^{p+1}} \right)^{\frac{1}{p}}}_{\coloneqq C} \Delta_{j-1}^{\frac{p+1}{p}}. \end{aligned} \tag{17}$$

This can be proved by investigating the univariate function $\vartheta(\tau) \coloneqq \tau - C \cdot \tau^{(p+1)/p}$. It is easy to verify that $\vartheta(\cdot)$ is nondecreasing in the region $[0, (p/((p+1)C))^p]$. Equation (15) guarantees that

$$\Delta_{j-1} \leq \frac{2\alpha H_{\max} R^{p+1}}{(p+1)!} \leq \left(\frac{p}{(p+1)C} \right)^p, \ \forall j \geq 2.$$

Since $F(x^{\mathcal{S}(j)-1}) - F(x^\star) \leq F(x^{\mathcal{S}(j-1)}) - F(x^\star)$, $\forall j \geq 2$, the univariate function $\vartheta(\cdot)$ satisfies

$$\vartheta(F(x^{\mathcal{S}(j)-1}) - F(x^\star)) \leq \vartheta(F(x^{\mathcal{S}(j-1)}) - F(x^\star)) = \vartheta(\Delta_{j-1}), \ \forall j \geq 2.$$

This is exactly the inequality from Equation (17). To this end, we derive the recursive inequality in terms of Δ_j from Equation (16):

$$\Delta_{j-1} - \Delta_j \geq C \cdot \Delta_{j-1}^{\frac{p+1}{p}}, \ \forall j \geq 2.$$

By a similar argument to the proof of Theorem 2 in Nesterov (2021), we conclude

$$\Delta_j \leq C^{-p}\left(\frac{1}{C\Delta_1^{1/p}}+\frac{j-1}{p}\right)^{-p} \leq C^{-p}\left(\frac{(p+1)^{(p+1)/p}}{p}+\frac{j-1}{p}\right)^{-p},\ \forall j\geq 1,$$

which implies

$$F(x^{\mathcal{S}(j)})-F(x^\star)\leq \frac{2\alpha H_{\max}R^{p+1}}{p!}\left(\frac{p+1}{j}\right)^p,\ \forall j\geq 1. \tag{18}$$

Note that for any iteration count k, it holds that $k=|\mathcal{S}\cap\{1,\dots,k\}|+|\mathcal{U}\cap\{1,\dots,k\}|$. Let $j(k):=|\mathcal{S}\cap\{1,\dots,k\}|$. Then for any $k\geq\overline{\mathcal{U}}$, we see

$$j(k)=k-|\mathcal{U}\cap\{1,\dots,k\}|\geq k-|\mathcal{U}|\geq k-\overline{\mathcal{U}}.$$

Since the method is monotone, we have $F(x^k)\leq F(x^{\mathcal{S}(j(k))})$, which implies

$$F(x^k)-F(x^\star)\leq \frac{2\alpha H_{\max}R^{p+1}}{p!}\left(\frac{p+1}{j(k)}\right)^p\leq \frac{2\alpha H_{\max}R^{p+1}}{p!}\left(\frac{p+1}{k-\overline{\mathcal{U}}}\right)^p,\ \forall k\geq\overline{\mathcal{U}}.$$

The proof is completed. □

Corollary 3.1. *Suppose the function f is convex and satisfies (3). Furthermore, assume that the initial sublevel set $\mathcal{F}(x^0):=\{x\in\mathrm{dom}(\psi)\mid F(x)\leq F(x^0)\}$ is bounded, i.e., $R:=\sup_{x,y\in\mathcal{F}(x^0)}\|x-y\|<\infty$. Let the sequence $\{x^k\}_{k\geq 0}$ be generated by Algorithm 1. Then for any accuracy $\epsilon>0$, Algorithm 1 requires at most*

$$K=\overline{\mathcal{U}}+\mathcal{O}\left(\left(\frac{\alpha H_{\max}R^{p+1}}{\epsilon}\right)^{\frac{1}{p}}\right)$$

iterations to return an iterate x^K such that $F(x^K)-F(x^\star)\leq\epsilon$.

Complexity analysis for the nonconvex case. We now turn to the case where the differentiable part f is nonconvex. Since the objective function F in problem (2) is nonsmooth, we measure the stationarity of a point $x\in\mathrm{dom}(\psi)$ by the distance $\mathrm{dist}(0,\partial F(x))$. Recalling the function value decrease from (11), the analysis reduces to relating the magnitude of the search direction d^k and to the stationarity measure $\mathrm{dist}(0,\partial F(x^{k-0.5}))$.

Lemma 3.2. *Suppose that function f satisfies (3). For any successful iteration, it holds that*

$$\mathrm{dist}(0,\partial F(x^{\mathcal{S}(j)}))\leq\frac{2\alpha H_{\max}}{p!}\|d^{\mathcal{S}(j)}\|^p,\ \forall j\geq 1.$$

Furthermore, if the nonsmooth term vanishes, i.e, $\psi\equiv 0$, it also holds that

$$-\lambda_{\min}\big(\nabla^2F(x^{\mathcal{S}(j)})\big)\leq\frac{2\alpha H_{\max}}{(p-1)!}\|d^{\mathcal{S}(j)}\|^{p-1},\ \forall j\geq 1.$$

Proof. Let $j\geq 1$. By definition of successful iterations, the function value decreases at iteration $\mathcal{S}(j)$ and hence $x^{\mathcal{S}(j)}=x^{\mathcal{S}(j)-1}+d^{\mathcal{S}(j)}$. By the first-order optimality of subproblem (8), we have

$$\nabla\Omega_p^{\sigma_{\mathcal{S}(j)}}(x^{\mathcal{S}(j)};x^{\mathcal{S}(j)-1})+\psi'(x^{\mathcal{S}(j)})=0,$$

where $\psi'(x^{\mathcal{S}(j)}) \in \partial\psi(x^{\mathcal{S}(j)})$. Note that

$$\begin{aligned}
&\nabla f(x^{\mathcal{S}(j)}) + \psi'(x^{\mathcal{S}(j)}) \\
&= \nabla f(x^{\mathcal{S}(j)}) - \nabla \Omega_p^{\sigma_{\mathcal{S}(j)}}(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) + \nabla \Omega_p^{\sigma_{\mathcal{S}(j)}}(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) + \psi'(x^{\mathcal{S}(j)}) \\
&= \nabla f(x^{\mathcal{S}(j)}) - \nabla \, \mathrm{T}_p(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) - \frac{\sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^{p-1} d^{\mathcal{S}(j)}.
\end{aligned}$$

Combining the above equation with inequality (5), we obtain

$$\begin{aligned}
&\|\nabla f(x^{\mathcal{S}(j)}) + \psi'(x^{\mathcal{S}(j)})\| \\
&\leq \|\nabla f(x^{\mathcal{S}(j)}) - \nabla \, \mathrm{T}_p(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1})\| + \frac{\sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^p \\
&\leq \frac{L_p + \sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^p \\
&\leq \frac{2\alpha H_{\max}}{p!} \|d^{\mathcal{S}(j)}\|^p.
\end{aligned}$$

We use inequalities $\sigma_{\mathcal{S}(j)} = \alpha M_{\mathcal{S}(j)} \leq \alpha H_{\max}$ and $L_p \leq H_{\max} \leq \alpha H_{\max}$ in the last step. As a result, by applying the relationship $\mathrm{dist}(0, x^{\mathcal{S}(j)}) \leq \|\nabla f(x^{\mathcal{S}(j)}) + \psi'(x^{\mathcal{S}(j)})\|$, we have $\mathrm{dist}(0, \partial F(x^{\mathcal{S}(j)})) \leq 2\alpha L_p \|d^{\mathcal{S}(j)}\|^p/(p!)$. Now we consider the case that $\psi \equiv 0$. First note that

$$\begin{aligned}
&\nabla^2 \Omega_p^{\sigma_{\mathcal{S}(j)}}(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) \\
&= \nabla^2 \, \mathrm{T}_p(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) + \frac{\sigma_{\mathcal{S}(j)}}{p!} \big((p-1)\|d^{\mathcal{S}(j)}\|^{p-3} d^{\mathcal{S}(j)} (d^{\mathcal{S}(j)})^\top + \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n}\big).
\end{aligned}$$

By the second-order optimality of subproblem (8), we have

$$\nabla^2 \Omega_p^{\sigma_{\mathcal{S}(j)}}(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) \succeq 0,$$

which implies

$$\nabla^2 \, \mathrm{T}_p(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) \succeq -\frac{\sigma_{\mathcal{S}(j)}}{p!} \big((p-1)\|d^{\mathcal{S}(j)}\|^{p-3} d^{\mathcal{S}(j)} (d^{\mathcal{S}(j)})^\top + \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n}\big).$$

By inequality (6), it follows

$$\begin{aligned}
&\nabla^2 F(x^{\mathcal{S}(j)}) \\
&\succeq \nabla^2 \, \mathrm{T}_p(x^{\mathcal{S}(j)}; x^{\mathcal{S}(j)-1}) - \frac{L_p}{(p-1)!} \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n} \\
&\succeq -\frac{\sigma_{\mathcal{S}(j)}}{p!} \big((p-1)\|d^{\mathcal{S}(j)}\|^{p-3} d^{\mathcal{S}(j)} (d^{\mathcal{S}(j)})^\top + \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n}\big) - \frac{L_p}{(p-1)!} \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n} \\
&= -\frac{(p-1)\sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^{p-3} d^{\mathcal{S}(j)} (d^{\mathcal{S}(j)})^\top - \frac{\sigma_{\mathcal{S}(j)} + pL_p}{p!} \|d^{\mathcal{S}(j)}\|^{p-1} \, \mathrm{I}_{n\times n}.
\end{aligned}$$

Therefore, we conclude

$$\begin{aligned}
\lambda_{\min}\big(\nabla^2 F(x^{\mathcal{S}(j)})\big) &\geq -\frac{(p-1)\sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^{p-3} \lambda_{\max}\big(d^{\mathcal{S}(j)} (d^{\mathcal{S}(j)})^\top\big) - \frac{\sigma_{\mathcal{S}(j)} + pL_p}{p!} \|d^{\mathcal{S}(j)}\|^{p-1} \\
&= -\frac{(p-1)\sigma_{\mathcal{S}(j)}}{p!} \|d^{\mathcal{S}(j)}\|^{p-1} - \frac{\sigma_{\mathcal{S}(j)} + pL_p}{p!} \|d^{\mathcal{S}(j)}\|^{p-1} \\
&= -\frac{\sigma_{\mathcal{S}(j)} + L_p}{(p-1)!} \|d^{\mathcal{S}(j)}\|^{p-1}.
\end{aligned}$$

The proof is completed by using inequalities $\sigma_{\mathcal{S}(j)} = \alpha M_{\mathcal{S}(j)} \leq \alpha H_{\max}$ and $L_p \leq H_{\max} \leq \alpha H_{\max}$ again. □

Theorem 3.2. *Suppose that function f satisfies (3). Let the sequence $\{x^k\}_{k\geq 0}$ be generated by Algorithm 1. Then for any accuracies $\epsilon_g, \epsilon_H > 0$, the algorithm requires at most*

$$K = \overline{\mathcal{U}} + \mathcal{O}\left(\frac{F(x^0) - F(x^\star)}{(\alpha-1)H_0} \cdot \max\left\{\left(\frac{\alpha H_{\max}}{\epsilon_g}\right)^{\frac{p+1}{p}}, \left(\frac{\alpha H_{\max}}{\epsilon_H}\right)^{\frac{p+1}{p-1}}\right\}\right)$$

iterations to return an iterate $\tilde{x}$ such that $\operatorname{dist}(0, \partial F(\tilde{x})) \leq \epsilon_g$. Furthermore, if the nonsmooth term vanishes, i.e, $\psi \equiv 0$, the sequence additionally satisfies $\lambda_{\min}(\nabla^2 F(\tilde{x})) \geq -\epsilon_H$.

Proof. Recall the definition of the successful index set in (12). Let $j \geq 1$. We have

$$F(x^{\mathcal{S}(j)}) - F(x^{\mathcal{S}(j)-1}) \leq -\frac{(\alpha-1)M_{\mathcal{S}(j)}}{2(p+1)!}\|d^{\mathcal{S}(j)}\|^{p+1}.$$

Summing over j from 1 to J implies

$$\sum_{j=1}^{J} \frac{(\alpha-1)M_{\mathcal{S}(j)}}{2(p+1)!}\|d^{\mathcal{S}(j)}\|^{p+1} \leq \sum_{j=1}^{J} F(x^{\mathcal{S}(j)-1}) - F(x^{\mathcal{S}(j)}) \leq \sum_{k=1}^{\mathcal{S}(J)} F(x^{k-1}) - F(x^k).$$

where we use the monotonicity of Algorithm 1. Since $M_{\mathcal{S}(j)} \geq H_0$, $\forall j \geq 1$, the above inequality follows

$$\sum_{j=1}^{J} \frac{(\alpha-1)H_0}{2(p+1)!}\|d^{\mathcal{S}(j)}\|^{p+1} \leq F(x^0) - F(x^\star),$$

which yields

$$\|d^{\mathcal{S}(j(\min))}\|^{p+1} := \min_{j=1,\ldots,J} \|d^{\mathcal{S}(j)}\|^{p+1} \leq \frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0 J}. \tag{19}$$

Clearly, $\mathcal{S}(j(\min)) \in \{\mathcal{S}(1), \ldots, \mathcal{S}(J)\}$. Let $K = \mathcal{S}(J)$. Note that $K = J + |\mathcal{U} \cap \{1, \ldots, K\}| \leq J + \overline{\mathcal{U}}$. The above inequality follows

$$\|d^{\mathcal{S}(j(\min))}\|^{p+1} \leq \frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0(K - \overline{\mathcal{U}})}.$$

Now we use the results in Lemma 3.2 to get

$$\operatorname{dist}(0, \partial F(x^{\mathcal{S}(j(\min))})) \leq \frac{2\alpha H_{\max}}{p!}\left(\frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0(K - \overline{\mathcal{U}})}\right)^{\frac{p}{p+1}}.$$

Therefore, after at most

$$K = \overline{\mathcal{U}} + \left\lceil \frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0}\left(\frac{2\alpha H_{\max}}{p!\epsilon_g}\right)^{\frac{p+1}{p}} \right\rceil$$

iterations, it holds that $\operatorname{dist}(0, \partial F(x^{\mathcal{S}(j(\min))})) \leq \epsilon_g$. In the special case where $\psi \equiv 0$ holds, from Lemma 3.2, we also have

$$-\lambda_{\min}\big(\nabla^2 F(x^{\mathcal{S}(j(\min))})\big) \leq \frac{2\alpha H_{\max}}{(p-1)!}\left(\frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0(K - \overline{\mathcal{U}})}\right)^{\frac{p-1}{p+1}}.$$

Similarly, after at most

$$K = \overline{\mathcal{U}} + \left\lceil \frac{2(p+1)!(F(x^0) - F(x^\star))}{(\alpha-1)H_0}\left(\frac{2\alpha H_{\max}}{(p-1)!\epsilon_H}\right)^{\frac{p+1}{p-1}} \right\rceil$$

iterations, it holds that $\lambda_{\min}\big(\nabla^2 F(x^{\mathcal{S}(j(\min))})\big) \geq -\epsilon_H$. The proof is completed. □

4 Practical Enhancements

In Algorithm 1, the adaptive parameter M_k is updated using information from all past iterates. This strategy ensures that M_k serves as a nondecreasing and close estimate of the true global Lipschitz constant L_p, which is key to guaranteeing convergence theoretically. However, this approach can be inefficient at times in practice. The global Lipschitz constant is often a conservative estimate, as the objective function may exhibit a much smaller local Lipschitz constant in certain regions. In such cases, relying on the global estimate prevents `HAR` from taking more aggressive steps. Besides, the maximum of all historical estimates rule is sensitive to outliers. If `HAR` encounters a single ill-conditioned point at some iteration k, the local Lipschitz estimate H_k can become excessively large. Since the adaptive parameter M_k never decreases, this single event can cause the regularization parameter to stagnate at a large value, making all subsequent steps conservative.

A remedy is to introduce a predefined budget to limit the number of historical estimates the algorithm stores. Specifically, we maintain a history of at most $\mathcal{B}$ local estimates, allowing `HAR` to periodically forget older information. This mechanism prevents a single large estimate from an ill-conditioned iterate and enables the scheme to adapt to regions where the local Lipschitz constant may be smaller. This motivates our cyclic history-aware adaptive regularization (`HAR-C`) method. In this scheme, we maintain a history of local estimates up to a budget of size $\mathcal{B}$. Once the number of stored local estimates reaches this budget, we perform a refresh by clearing the history. We reset the adaptive parameter M_k to the maximum of its initial guess and the current local estimate, thereby discarding outdated information as the algorithm progresses toward the optimum.

Algorithm 2: Cyclic History-Aware Adaptive Regularization Method

Input: Initial point $x^0 \in \mathrm{dom}(\psi)$, initial guess $H_0 = M_0 > 0$, parameter $\alpha > 1$, budget parameter $\mathcal{B} \in \mathbb{N} \cup \{\infty\}$
For $k = 1, 2, \ldots$ **do**
  If $k \bmod \mathcal{B} = 0$ **then**
    Refresh the adaptive parameter $M_k = \max\{H_0, H_{k-1}\}$;
  Else
    Update the adaptive parameter $M_k = \max\{M_{k-1}, H_{k-1}\}$;
  End If
  Calculate the regularization parameter $\sigma_k = \alpha M_k$;
  Compute the search direction

$$d^k = \mathrm{argmin}_{d \in \mathrm{dom}(\psi)} \, \Omega_p^{\sigma_k}(x^{k-1} + d; x^{k-1}) + \psi(x^{k-1} + d);$$

  Update the intermediate iterate $x^{k-0.5} = x^{k-1} + d^k$;
  Update the iterate $x^k = \mathrm{argmin}_{x \in \{x^{k-0.5}, x^{k-1}\}} F(x)$;
  Update the local Lipschitz estimate

$$\frac{H_k}{(p+1)!} = \frac{f(x^{k-0.5}) - \mathrm{T}_p(x^{k-1} + d^k; x^{k-1})}{\|d^k\|^{p+1}};$$

End For

From the analysis of `HAR` in Section 3, a key element is the finiteness of the unsuccessful index set. This property and the monotone step together ensure that function value decrease is established during the successful iterations, even though they may be interspersed with unsuccessful ones.

Intuitively, to ensure convergence, the budget $\mathcal{B}$ in HAR-C should be larger than $\overline{\mathcal{U}}$. The trade-off that naturally arises is that HAR-C is no longer parameter-free, as this requirement necessitates a rough upper estimate of the global Lipschitz constant. Let $\overline{L_p}$ be an upper bound of the true Lipschitz constant L_p. If the budget $\mathcal{B}$ satisfies the following condition:

$$\mathcal{B} \geq \left\lceil \log_{\frac{\alpha+1}{2}} \frac{\max\{\overline{L_p}, H_0\}}{H_0} \right\rceil + 1, \tag{20}$$

then HAR-C is guaranteed to not get stuck. Recall that in the proofs of Theorems 3.1 and 3.2, we first establish the convergence behavior in terms of the successful iterates (see Equation (18) and Lemma 3.2), and then translate this behavior to the entire sequence of iterates. Note that the convergence behavior of the successful iterates is not affected by the cyclic update introduced in HAR-C. Therefore, the key difference compared with HAR is the bound on the number of null iterations. In HAR-C, up to $\overline{\mathcal{U}}$ unsuccessful iterations can occur in each cycle.

Lemma 4.1. *Suppose that function f satisfies (3). The budget satisfies condition (20). The cardinality of the unsuccessful index set $\mathcal{U}$ in Algorithm 2 satisfies*

$$|\mathcal{U} \cap \{1, \ldots, k\}| \leq \left\lceil \frac{k}{\mathcal{B}} \right\rceil \overline{\mathcal{U}}, \ \forall k \geq 1.$$

Proof. Since the budget $\mathcal{B}$ satisfies condition (20), by Lemma 3.1, there exists at least one successful iteration in the cycle, and thus HAR-C does not get stuck. Let $N_{\text{cyc}}(k) = \lceil k/\mathcal{B} \rceil$ be the current cycle number at iteration k. From Lemma 3.1, we know that at most $\overline{\mathcal{U}}$ unsuccessful iterations can occur within each cycle. Therefore, the total number of unsuccessful iterations up to iteration k is upper bounded by $N_{\text{cyc}}(k)\overline{\mathcal{U}}$, which completes the proof. □

Theorem 4.1. *Suppose the function f is convex and satisfies (3). The budget satisfies condition (20). Furthermore, assume that the initial sublevel set $\mathcal{F}(x^0) := \{x \in \text{dom}(\psi) \mid F(x) \leq F(x^0)\}$ is bounded, i.e., $R := \sup_{x,y \in \mathcal{F}(x^0)} \|x - y\| < \infty$. Then the sequence $\{x^k\}_{k \geq 0}$ generated by Algorithm 2 satisfies*

$$F(x^k) - F(x^\star) \leq \frac{2\alpha H_{\max} R^{p+1}}{p!} \left(\frac{p+1}{k(1 - \overline{\mathcal{U}}/\mathcal{B}) - \overline{\mathcal{U}}} \right)^p, \ \forall k \geq \frac{\overline{\mathcal{U}}}{1 - \overline{\mathcal{U}}/\mathcal{B}}.$$

Proof. From Equation (18), we know the convergence behavior in terms of the successful iterates is

$$F(x^{\mathcal{S}(j)}) - F(x^\star) \leq \frac{2\alpha H_{\max} R^{p+1}}{p!} \left(\frac{p+1}{j} \right)^p, \ \forall j \geq 1.$$

Besides, note that for any iteration count k, it holds that $k = |\mathcal{S} \cap \{1, \ldots, k\}| + |\mathcal{U} \cap \{1, \ldots, k\}|$. Let $j(k) := |\mathcal{S} \cap \{1, \ldots, k\}|$. Then from Lemma 4.1, we have

$$j(k) = k - |\mathcal{U} \cap \{1, \ldots, k\}| \geq k - \left\lceil \frac{k}{\mathcal{B}} \right\rceil \overline{\mathcal{U}} \geq k \left(1 - \frac{\overline{\mathcal{U}}}{\mathcal{B}} \right) - \overline{\mathcal{U}}.$$

Note that $F(x^k) - F(x^\star) \leq F(x^{\mathcal{S}(j(k))}) - F(x^\star)$ due to the monotonicity of HAR-C. Therefore, we conclude

$$F(x^k) - F(x^\star) \leq \frac{2\alpha H_{\max} R^{p+1}}{p!} \left(\frac{p+1}{j(k)} \right)^p \leq \frac{2\alpha H_{\max} R^{p+1}}{p!} \left(\frac{p+1}{k(1 - \overline{\mathcal{U}}/\mathcal{B}) - \overline{\mathcal{U}}} \right)^p, \ \forall k \geq \frac{\overline{\mathcal{U}}}{1 - \overline{\mathcal{U}}/\mathcal{B}}.$$

The proof is completed. □

Corollary 4.1. *Suppose the function f is convex and satisfies (3). The budget satisfies condition (20). Furthermore, assume that the initial sublevel set $\mathcal{F}(x^0) \coloneqq \{x \in \mathrm{dom}(\psi) \mid F(x) \leq F(x^0)\}$ is bounded, i.e., $R \coloneqq \sup_{x,y\in\mathcal{F}(x^0)} \|x-y\| < \infty$. Let the sequence $\{x^k\}_{k\geq 0}$ be generated by Algorithm 1. Then for any accuracy $\epsilon > 0$, Algorithm 2 requires at most*

$$K = \frac{\overline{\mathcal{U}}}{1-\overline{\mathcal{U}}/\mathcal{B}} + \mathcal{O}\left(\frac{1}{1-\overline{\mathcal{U}}/\mathcal{B}}\left(\frac{\alpha H_{\max} R^{p+1}}{\epsilon}\right)^{\frac{1}{p}}\right)$$

iterations to return an iterate x^K such that $F(x^K) - F(x^\star) \leq \epsilon$.

Theorem 4.2. *Suppose that function f satisfies (3). The budget satisfies condition (20). Let the sequence $\{x^k\}_{k\geq 0}$ be generated by Algorithm 2. Then for any accuracies $\epsilon_g, \epsilon_H > 0$, the algorithm requires at most*

$$K = \frac{\overline{\mathcal{U}}}{1-\overline{\mathcal{U}}/\mathcal{B}} + \mathcal{O}\left(\frac{F(x^0)-F(x^\star)}{(\alpha-1)(1-\overline{\mathcal{U}}/\mathcal{B})H_0} \cdot \max\left\{\left(\frac{\alpha H_{\max}}{\epsilon_g}\right)^{\frac{p+1}{p}}, \left(\frac{\alpha H_{\max}}{\epsilon_H}\right)^{\frac{p+1}{p-1}}\right\}\right)$$

iterations to return an iterate $\tilde{x}$ such that $\mathrm{dist}(0, \partial F(\tilde{x})) \leq \epsilon_g$. Furthermore, if the nonsmooth term vanishes, i.e, $\psi \equiv 0$, the sequence additionally satisfies $\lambda_{\min}(\nabla^2 F(\tilde{x})) \geq -\epsilon_H$.

Proof. Since the introduced cycle does not affect the convergence behavior of the successful iterates, Equation (19) still holds. That is

$$\|d^{\mathcal{S}(j(\min))}\|^{p+1} \coloneqq \min_{j=1,\ldots,J} \|d^{\mathcal{S}(j)}\|^{p+1} \leq \frac{2(p+1)!(F(x^0)-F(x^\star))}{(\alpha-1)H_0 J}.$$

Clearly, $\mathcal{S}(j(\min)) \in \{\mathcal{S}(1), \ldots, \mathcal{S}(J)\}$. Let $K = \mathcal{S}(J)$. Again, we use Lemma 4.1 to get

$$K = J + |\mathcal{U} \cap \{1,\ldots,K\}| \leq J + \left\lceil \frac{K}{\mathcal{B}} \right\rceil \overline{\mathcal{U}} \leq J + \left(\frac{K}{\mathcal{B}} + 1\right)\overline{\mathcal{U}}.$$

This implies $J \geq K\left(1 - \frac{\overline{\mathcal{U}}}{\mathcal{B}}\right) - \overline{\mathcal{U}}$. Substituting this inequality back and using Lemma 3.2 completes the proof. □

As an alternative to the cyclic refresh strategy, we introduce the sliding-window history-aware adaptive regularization (`HAR-S`) method in Algorithm 3. This approach maintains a sliding window, using only the $\mathcal{B}$ most recent local estimates to update the adaptive parameter. Theoretically, `HAR-S` converges as long as the choice of $\mathcal{B}$ satisfies the condition (20). We first establish an upper bound on the cardinality of unsuccessful iterates in `HAR-S` that is identical to the one in Lemma 4.1.

Lemma 4.2. *Suppose that function f satisfies (3). The budget satisfies condition (20). The cardinality of the unsuccessful index set $\mathcal{U}$ in Algorithm 3 satisfies*

$$|\mathcal{U} \cap \{1,\ldots,k\}| \leq \left\lceil \frac{k}{\mathcal{B}} \right\rceil \overline{\mathcal{U}}, \ \forall k \geq 1.$$

Proof. We first claim that for any window $W \subseteq \mathbb{N}$ with length $\mathcal{B}$, it contains at most $\overline{\mathcal{U}}$ unsuccessful iterations. Choose two consecutive unsuccessful iterations $\mathcal{U}(j), \mathcal{U}(j+1) \in W$. Then it holds that $\mathcal{U}(j+1) - \mathcal{U}(j) \leq \mathcal{B} - 1$, which implies

$$\mathcal{U}(j) \in [\mathcal{U}(j+1) - \mathcal{B} + 1, \mathcal{U}(j+1) - 1].$$

Algorithm 3: Sliding-Window History-Aware Adaptive Regularization Method

Input: Initial point $x^0 \in \mathrm{dom}(\psi)$, initial guess $H_0 = M_0 > 0$, parameter $\alpha > 1$, budget parameter $\mathcal{B} \in \mathbb{N} \cup \{\infty\}$

For $k = 1, 2, \ldots$ **do**

Calculate $b_k = \max\{1, k - \mathcal{B}\}$;

Update the adaptive parameter $M_k = \max\{H_0, H_{b_k}, H_{b_k+1}, \ldots, H_{k-1}\}$;

Calculate the regularization parameter $\sigma_k = \alpha M_k$;

Compute the search direction

$$d^k = \mathrm{argmin}_{d \in \mathrm{dom}(\psi)} \, \Omega_p^{\sigma_k}(x^{k-1} + d; x^{k-1}) + \psi(x^{k-1} + d);$$

Update the intermediate iterate $x^{k-0.5} = x^{k-1} + d^k$;

Update the iterate $x^k = \mathrm{argmin}_{x \in \{x^{k-0.5}, x^{k-1}\}} F(x)$;

Update the local Lipschitz estimate

$$\frac{H_k}{(p+1)!} = \frac{f(x^{k-0.5}) - \mathrm{T}_p(x^{k-1} + d^k; x^{k-1})}{\|d^k\|^{p+1}};$$

End For

Since $b_{\mathcal{U}(j+1)} = \max\{1, \mathcal{U}(j+1) - \mathcal{B}\} \leq \mathcal{U}(j+1) - \mathcal{B} + 1$, we have the interval inclusion

$$[\mathcal{U}(j+1) - \mathcal{B} + 1, \mathcal{U}(j+1) - 1] \subseteq [b_{\mathcal{U}(j+1)}, \mathcal{U}(j+1) - 1].$$

This yields $\mathcal{U}(j) \in [b_{\mathcal{U}(j+1)}, \mathcal{U}(j+1) - 1]$, and thus $H_{\mathcal{U}(j)} \leq M_{\mathcal{U}(j+1)}$ follows. Besides, $H_{\mathcal{U}(j+1)} > (\alpha+1) M_{\mathcal{U}(j+1)}/2$ holds since $\mathcal{U}(j+1)$ is an unsuccessful iteration. As a result, we obtain $H_{\mathcal{U}(j+1)} > (\alpha+1) H_{\mathcal{U}(j)}/2$. Suppose there exists J unsuccessful iterations inside window W. Without loss of generality, order these unsuccessful iterations as $\mathcal{U}(j) < \mathcal{U}(j+1) < \ldots < \mathcal{U}(j+J-1)$. By recursively using the previous inequality, we have

$$H_{\mathcal{U}(j+J-1)} \geq \left(\frac{\alpha+1}{2}\right)^{J-1} H_{\mathcal{U}(j)}.$$

Similarly, since $\mathcal{U}(j)$ is an unsuccessful iteration, it also holds that $H_{\mathcal{U}(j)} > (\alpha+1) M_{\mathcal{U}(j)}/2 \geq (\alpha+1) H_0/2$. Consequently, we obtain

$$H_{\mathcal{U}(j+J-1)} \geq \left(\frac{\alpha+1}{2}\right)^{J} H_0.$$

Clearly, it follows that $J \leq \overline{\mathcal{U}}$. Partition $\{1, \ldots, k\}$ into $m \coloneqq \lceil k/\mathcal{B} \rceil$ disjoint blocks

$$\mathcal{J}_t \coloneqq \{(t-1)\mathcal{B} + 1, \ldots, \min\{t\mathcal{B}, k\}\}, \; t = 1, \ldots, m,$$

each of length is smaller than $\mathcal{B}$. Therefore, summing over the disjoint blocks implies

$$|\mathcal{U} \cap \{1, \ldots, k\}| = \sum_{t=1}^{m} |\mathcal{U} \cap \mathcal{J}_t| \leq \sum_{t=1}^{m} \overline{\mathcal{U}} = \left\lceil \frac{k}{\mathcal{B}} \right\rceil \overline{\mathcal{U}}.$$

The proof is completed. □

Based on the above lemma, one can establish the convergence results of HAR-S by using arguments similar to those in Theorems 4.1 and 4.2. For completeness, we provide the results as follows; the proof is omitted.

Theorem 4.3. *Suppose the function f is convex and satisfies (3). The budget satisfies condition (20). Furthermore, assume that the initial sublevel set $\mathcal{F}(x^0) := \{x \in \mathrm{dom}(\psi) \mid F(x) \le F(x^0)\}$ is bounded, i.e., $R := \sup_{x,y\in\mathcal{F}(x^0)} \|x - y\| < \infty$. Then the sequence $\{x^k\}_{k\ge 0}$ generated by Algorithm 3 satisfies*

$$F(x^k) - F(x^\star) \le \frac{2\alpha H_{\max} R^{p+1}}{p!} \left(\frac{p+1}{k(1-\overline{\mathcal{U}}/\mathcal{B}) - \overline{\mathcal{U}}} \right)^p, \ \forall k \ge \frac{\overline{\mathcal{U}}}{1-\overline{\mathcal{U}}/\mathcal{B}}.$$

Theorem 4.4. *Suppose that function f satisfies (3). The budget satisfies condition (20). Let the sequence $\{x^k\}_{k\ge 0}$ be generated by Algorithm 3. Then for any accuracies $\epsilon_g, \epsilon_H > 0$, the algorithm requires at most*

$$K = \frac{\overline{\mathcal{U}}}{1-\overline{\mathcal{U}}/\mathcal{B}} + \mathcal{O}\left(\frac{F(x^0) - F(x^\star)}{(\alpha-1)(1-\overline{\mathcal{U}}/\mathcal{B})H_0} \cdot \max\left\{ \left(\frac{\alpha H_{\max}}{\epsilon_g}\right)^{\frac{p+1}{p}}, \left(\frac{\alpha H_{\max}}{\epsilon_H}\right)^{\frac{p+1}{p-1}} \right\} \right)$$

iterations to return an iterate $\tilde{x}$ such that $\mathrm{dist}(0, \partial F(\tilde{x})) \le \epsilon_g$. Furthermore, if the nonsmooth term vanishes, i.e, $\psi \equiv 0$, the sequence additionally satisfies $\lambda_{\min}(\nabla^2 F(\tilde{x})) \ge -\epsilon_H$.

Remark 4.1. *Note that when $\mathcal{B} = \infty$, both methods HAR-C and HAR-S reduce to HAR. In this case, the iteration complexity results established above match those in Theorem 3.1 and Corollary 3.1. While introducing the budget is expected to worsen the worst-case complexity guarantee, the numerical experiments in Section 6 show the practical superiority of using partial historical information.*

5 Acceleration for Convex Objectives

In this section, we develop an accelerated history-aware adaptive regularization method (HAR-A), presented in Algorithm 4, for convex objectives. The overall algorithm design is motivated by the original accelerated pth-order tensor method proposed in Nesterov (2021). We maintain four sequences in HAR-A: $\{x^k\}_{k\ge 0}$, $\{v^k\}_{k\ge 0}$, and $\{y^k\}_{k\ge 0}$ are the sequences of iterates, and $\{s^k\}_{k\ge 0}$ is used to accumulate past derivative information. It is well-known that Nesterov's acceleration can render optimization methods non-monotone (e.g., the Nesterov accelerated gradient method). Therefore, in the development of HAR-A, we introduce a repeat mechanism (Line 5 of Algorithm 4) to ensure that all sequences remain properly synchronized and updated.

The theoretical analysis comes from a modified estimating sequence technique in Nesterov (2008). The global convergence of HAR-A is guaranteed by maintaining the following two relations across iterations

$$\phi_k^\star \ge A_k F(x^k), \ \phi_k^\star := \min_{x\in\mathrm{dom}(\psi)} \phi_k(x), \tag{21a}$$

$$\phi_k(x) \le A_k F(x) + M_k C_p(\alpha,\beta)\|x - x^0\|^{p+1}, \ \forall x \in \mathrm{dom}(\psi). \tag{21b}$$

Here $\{\phi_k(x)\}_{k\ge 0}$ is a sequence of functions that approximate $F(x)$ from both above and below. As we will show later, v^k is the optimal point of $\phi_k(\cdot)$, i.e., $\phi_k(v^k) = \phi_k^\star$. Besides, $\{A_k\}_{k\ge 0}$ is a sequence that measures the convergence rate of the sequence $\{x^k\}_{k\ge 0}$. As a direct result of relationship (21), we have

$$F(x^k) - F(x^\star) \le \frac{M_k C_p(\alpha,\beta)\|x^0 - x^\star\|^{p+1}}{A_k}, \ \forall k \ge 0.$$

Algorithm 4: Accelerated History-Aware Adaptive Regularization Method

Input: Initial point $x^0 = v^0 \in \mathrm{dom}(\psi)$, $s^0 = 0$, initial guess $H_0 = M_0 > 0$, parameter $\alpha > 1$
Initialization: set $\beta = (\alpha+1)/2$, set parameters $C(\alpha,\beta)$ as in (23) and A_k, a_k as in (22)
For $k = 0, 1, \ldots$ **do**
  Update the iterate $y^k = \frac{A_k}{A_k+a_k}x^k + \frac{a_k}{A_k+a_k}v^k$;
  Repeat
    Calculate the regularization parameter $\sigma_k = \alpha M_k$;
    Compute the search direction

$$d^k = \mathrm{argmin}_{d\in\mathrm{dom}(\psi)}\, \Omega_p^{\sigma_k}(y^k + d; y^k) + \psi(y^k + d);$$

    Update the iterate $x^{k+1} = y^k + d^k$;
    Set the subgradient $g^{k+1} := \nabla f(x^{k+1}) - \nabla\Omega_p^{\sigma_k}(y^k + d^k; y^k) \in \partial F(x^{k+1})$;
    Update the local Lipschitz estimate

$$\frac{H_{k+1}}{p!} = \frac{\|g^{k+1} + \frac{\sigma_k}{p!}\|d^k\|^{p-1}d^k\|}{\|d^k\|^p};$$

    If $H_{k+1} \geq \beta M_k$ **then**
      Update the adaptive parameter $M_k = H_{k+1}$;
    Else
      Break;
    End If;
  Update the adaptive parameter $M_{k+1} = \max\{M_k, H_{k+1}\}$;
  Update $s^{k+1} = s^k + a_k g^{k+1}$;
  Update the iterate $v^{k+1} = v^0 - \big((p+1)C_p(\alpha,\beta)M_{k+1}\big)^{-1/p}\|s^{k+1}\|^{-(p-1)/p}s^{k+1}$;
End For

Therefore, the global complexity directly follows from the choice of A_k and M_k. For the upcoming analysis, we first complete the definition of $A_k, \phi_k(x)$ in the estimating sequence framework as follows:

$$A_k = k^{p+1},\ a_k = A_{k+1} - A_k = (k+1)^{p+1} - k^{p+1}, \tag{22}$$

$$C_p(\alpha,\beta) = \frac{(p+1)^{(3p+1)/2}(p-1)^{(3p-1)/2}}{2^p(\alpha^2-\beta^2)^{(p-1)/2}}, \tag{23}$$

$$\begin{aligned}\phi_0(x) &= M_0 C_p(\alpha,\beta)\|x - x^0\|^{p+1},\\ \phi_{k+1}(x) &= \phi_k(x) + a_k\Big(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\Big)\\ &\quad + C_p(\alpha,\beta)(M_{k+1} - M_k)\|x - x^0\|^{p+1}.\end{aligned} \tag{24}$$

We begin with some basic properties of the estimating sequence, which are modified from Section 3 in Nesterov (2021).

Lemma 5.1. *For any $k \geq 0$, the sequences $\{A_k\}_{k\geq 0}$ and $\{a_k\}_{k\geq 0}$ satisfy*

$$A_{k+1}^{-1}a_k^{(p+1)/p} \leq (p+1)^{(p+1)/p}. \tag{25}$$

Lemma 5.2. *For any $k \geq 0$, the iterate v^k is the unique minimizer of $\phi_k(x)$. Besides, it holds that*

$$\phi_k(x) \geq \phi_k^\star + \frac{M_k C_p(\alpha,\beta)}{2^{p-1}}\|x - v^k\|^{p+1}. \tag{26}$$

Proof. First note that

$$\phi_k(x) = C_p(\alpha,\beta)M_k\|x - x^0\|^{p+1} + \sum_{i=0}^{k-1} a_i \left(F(x^{i+1}) + \langle \nabla g_{i+1}, x - x^{i+1}\rangle\right). \tag{27}$$

Then from the optimality condition, we have

$$\begin{aligned} 0 = \nabla\phi_k(x) &= \nabla\left(C_p(\alpha,\beta)M_k\|x - x^0\|^{p+1}\right) + \sum_{i=0}^{k-1} a_i g_{i+1} \\ &= (p+1)C_p(\alpha,\beta)M_k\|x - x^0\|^{p-1}(x - x^0) + \sum_{i=0}^{k-1} a_i g_{i+1}. \end{aligned}$$

Solving the optimality condition and observing the update rule for s^k yields

$$x = v^0 - \frac{1}{\left[(p+1)C_p(\alpha,\beta)M_k\right]^{1/p}\|s_k\|^{(p-1)/p}} s_k,$$

which is precisely v^k as defined in Line 18 of Algorithm 4. Finally, Equation (26) comes from the fact that $\phi_k(x)$ is uniformly convex of degree p. □

Lemma 5.3. *Suppose f satisfies (3). Then during the whole process of Algorithm 4, the total number of repetitions at each iteration is bounded by $\left\lceil \log_\beta \frac{H_{\max}}{H_0} \right\rceil$. Therefore, H_k and M_k has an upper bound $H_{\max}$.*

Proof. The proof is identical to the proof of Lemma 3.1. □

Now we demonstrate that the generated search direction d_k in `HAR-A` satisfies the following property, which is essential for acceleration (see Corollary 1 in Nesterov (2021)).

Lemma 5.4. *Suppose f is convex and satisfies (3). Then for each iteration of Algorithm 4, it holds that*

$$\langle y^k - x^{k+1}, g^{k+1}\rangle > D(\alpha,\beta)\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}}, \ k \geq 0, \tag{28}$$

where the constant $D(\alpha,\beta) \coloneqq (\alpha^2 - \beta^2)^{(p-1)/2p}\left(\frac{2p}{p-1}\right)\left(\frac{p-1}{p+1}\right)^{(p+1/2p)}$.

Proof. Note that $\frac{H_{k+1}}{p!} = \frac{\|g^{k+1} + \frac{\sigma_k}{p!}\|d^k\|^{p-1}d^k\|}{\|d^k\|^p}$, and the mechanism guarantees that $H_{k+1} < \beta M_k$ in the end. Squaring both sides of the former equation gives

$$\frac{H_{k+1}^2}{(p!)^2}\|d^k\|^{2p} = \frac{\alpha^2 M_k^2}{(p!)^2}\|d^k\|^{2p} + \frac{2}{p!}\sigma_k\|d^k\|^{p-1}\langle d^k, g^{k+1}\rangle + \|g^{k+1}\|^2.$$

Then by arranging terms, we have

$$\begin{aligned}\langle y^k - x^{k+1}, g^{k+1}\rangle &= \frac{p!}{2\sigma_k\|d^k\|^{p-1}}\left(\frac{\alpha^2 M_k^2 - H_{k+1}^2}{(p!)^2}\|d^k\|^{2p} + \|g^{k+1}\|^2\right)\\ &> \frac{p!}{2\sigma_k}\left(\frac{(\alpha^2-\beta^2)M_k^2}{(p!)^2}\|d^k\|^{p+1} + \frac{\|g^{k+1}\|^2}{\|d^k\|^{p-1}}\right).\end{aligned}$$

Introduce a univariate auxiliary function $h(t) \coloneqq \gamma t^{\frac{p+1}{p-1}} + \frac{\xi}{t},\ \gamma,\ \xi,\ t > 0$. It is easy to verify that

$$\min_{t>0} h(t) = \frac{2p}{p-1}\left(\frac{\xi(p-1)}{p+1}\right)^{\frac{p+1}{2p}}\gamma^{\frac{p-1}{2p}}.$$

By plugging in $\xi = \frac{p!\|g^{k+1}\|^2}{2\sigma_k}$ and $\gamma = \frac{(\alpha^2-\beta^2)M_k}{2\alpha p!}$, we obtai

$$\langle y^k - x^{k+1}, g^{k+1}\rangle > (\alpha^2-\beta^2)^{(p-1)/2p}\left(\frac{2p}{p-1}\right)\left(\frac{p-1}{p+1}\right)^{(p+1/2p)}\cdot\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}}.$$

The proof is completed. □

Lemma 5.5. *Suppose the function f is convex and satisfies (3). Then for each iteration of Algorithm 4, it holds that*

$$A_k F(x^k) \le \phi_k^\star \le \phi_k(x) \le A_k F(x) + M_k C_p(\alpha,\beta)\|x - x^0\|^{p+1},\ \forall x \in \mathrm{dom}(\psi),\ k \ge 0, \tag{29}$$

which is a compact form of Equation (21).

Proof. We prove by induction, note that $A_0 = 0$, so Equation (29) holds for the trivial case $i = 0$. Suppose it holds for the case $i = k$. We first check Equation (21b) for the case $i = k+1$. Note that

$$\begin{aligned}\phi_{k+1}(x) &= \phi_k(x) + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right) + C_p(\alpha,\beta)(M_{k+1}-M_k)\|x-x^0\|^{p+1}\\ &\le A_k F(x) + M_k C_p(\alpha,\beta)\|x-x^0\|^{p+1} + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right)\\ &\quad + C_p(\alpha,\beta)(M_{k+1}-M_k)\|x-x^0\|^{p+1}\\ &\le A_k F(x) + a_k F(x) + C_p(\alpha,\beta)M_{k+1}\|x-x^0\|^{p+1}\\ &= A_{k+1}F(x) + C_p(\alpha,\beta)M_{k+1}\|x-x^0\|^{p+1},\end{aligned}$$

where the first inequality is from the induction hypothesis. We now check Equation (21a) for the case $i = k+1$. First by the uniform convexity of ϕ_k and the induction hypothesis, it follows

$$\begin{aligned}\phi_{k+1}^\star &= \min_{x\in\mathrm{dom}(\psi)}\left\{\phi_k(x) + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right) + C_p(\alpha,\beta)(M_{k+1}-M_k)\|x-x^0\|^{p+1}\right\}\\ &\ge \min_{x\in\mathrm{dom}(\psi)}\left\{\phi_k(x) + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right)\right\}\\ &\ge \min_{x\in\mathrm{dom}(\psi)}\left\{\phi_k^\star + \frac{C_p(\alpha,\beta)M_k}{2^{p-1}}\|x-v^k\|^{p+1} + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right)\right\}\\ &\ge \min_{x\in\mathrm{dom}(\psi)}\left\{A_k F(x^k) + \frac{C_p(\alpha,\beta)M_k}{2^{p-1}}\|x-v^k\|^{p+1} + a_k\left(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\right)\right\}.\end{aligned}$$

Since F is convex, we immediately obtain

$$\begin{aligned}
&\phi_{k+1}^\star \\
&\geq \min_{x\in\mathrm{dom}(\psi)}\left\{A_kF(x^{k+1}) + A_k\langle g^{k+1}, x^k - x^{k+1}\rangle + \frac{C_p(\alpha,\beta)M_k}{2^{p-1}}\|x-v^k\|^{p+1} + a_k\Big(F(x^{k+1}) + \langle g^{k+1}, x - x^{k+1}\rangle\Big)\right\}
\end{aligned}$$

from the above equation. After some algebraic calculation, it holds that

$$\begin{aligned}
\phi_{k+1}^\star &\geq A_{k+1}F(x^{k+1}) + A_{k+1}\left\langle g^{k+1}, \frac{A_k}{A_{k+1}}x^k + \frac{a_k}{A_{k+1}}v^k - x^{k+1}\right\rangle \\
&\quad + \min_{x\in\mathrm{dom}(\psi)}\left\{\frac{C_p(\alpha,\beta)M_k}{2^{p-1}}\|x-v^k\|^{p+1} + a_k\langle g^{k+1}, x - v^k\rangle\right\} \\
&= A_{k+1}F(x^{k+1}) + A_{k+1}\left\langle g^{k+1}, \frac{A_k}{A_{k+1}}x^k + \frac{a_k}{A_{k+1}}v^k - x^{k+1}\right\rangle \\
&\quad - a_k^{(p+1)/p}\cdot\frac{p2^{(p-1)(p+1)/p}}{C_p^{1/p}(\alpha,\beta)(p+1)^{(p+1/p)}}\cdot\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}} \\
&= A_{k+1}F(x^{k+1}) + A_{k+1}\langle g^{k+1}, y^k - x^{k+1}\rangle - a_k^{(p+1)/p}\cdot\frac{p2^{(p-1)(p+1)/p}}{C_p^{1/p}(\alpha,\beta)(p+1)^{(p+1/p)}}\cdot\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}} \\
&= A_{k+1}F(x^{k+1}) + A_{k+1}\left(\langle g^{k+1}, y^k - x^{k+1}\rangle - A_{k+1}^{-1}a_k^{(p+1)/p}\cdot\frac{p2^{(p-1)(p+1)/p}}{C_p^{1/p}(\alpha,\beta)(p+1)^{(p+1/p)}}\cdot\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}}\right).
\end{aligned}$$

Finally, by using Equations (25) and (28), we concludde

$$\begin{aligned}
\phi_{k+1}^\star &\geq A_{k+1}F(x^{k+1}) + A_{k+1}\left(\langle g^{k+1}, y^k - x^{k+1}\rangle - D(\alpha,\beta)\cdot\frac{\|g^{k+1}\|^{(p+1)/p}}{M_k^{1/p}}\right) \\
&\geq A_{k+1}F(x^{k+1}).
\end{aligned}$$

The proof is completed. □

As a direct consequence of combining Equation (22), Lemma 5.3, and Lemma 5.5, we have the following iteration complexity result of Algorithm 4. Note that our result is established in terms of the function value gap. By incorporating `HAR-A` into the recently proposed accumulative regularization framework (Ji and Lan, 2025), the convergence guarantee can be translated to a guarantee on the gradient norm, which remains $\mathcal{O}(1/k^{p+1})$ and is still adaptive.

Theorem 5.1. *Suppose the function f is convex and satisfies (3). Then the sequence $\{x^k\}_{k\geq 0}$ generated by Algorithm 4 satisfies*

$$F(x^k) - F(x^\star) \leq \frac{H_{\max}C_p(\alpha,\beta)\|x^0 - x^\star\|^{p+1}}{k^{p+1}},\ \forall k \geq 0.$$

Remark 5.1. *The above iteration complexity $\mathcal{O}(1/k^{p+1})$ is not optimal when $p \geq 2$ compared with the lower bound established in Garg et al. (2021). However, to achieve the optimal iteration complexity, one generally requires some highly nontrivial line-search procedure (see Monteiro and Svaiter (2013); Kovalev and Gasnikov (2022); Carmon et al. (2022)), which is hard to implement. As a result, we leave the development of an easily implementable, adaptive optimal pth-order tensor method for future work.*

Remark 5.2. *It is known that accelerated methods often underperform in practice due to the lack of local convergence. Their stronger worst-case complexity guarantees often come at the expense of local efficiency. Empirical evidence indicates that in many problems, such as logistic regression, accelerated methods cannot match unaccelerated counterparts when searching for solutions in the high-precision regimes; see Jiang et al. (2020); Huang et al. (2024); Chen et al. (2022); Carmon et al. (2022). Consequently, it was often necessary to manually switch to Newton's method to achieve high-accuracy solutions.*

6 Numerical Experiments

We implement two practical history-aware methods using second-order derivatives, i.e., $p = 2$, and compare them to the adaptive cubic regularized Newton method. In regard of Remark 5.2, we do not provide an implementation of Algorithm 4.

- `HAR-C` (Algorithm 2) and `HAR-S` (Algorithm 3), each tested with different budget sizes $\mathcal{B}$ (cf. (20)). The resulting cubic subproblems are solved using the textbook one-dimensional line-search strategy as described in Nesterov and Polyak (2006), endowed with Cholesky factorization to solve the arising linear systems. Because these subproblems are solved through direct factorizations, we do not employ inexact termination techniques commonly used in iterative Krylov solvers; see, e.g., Curtis et al. (2021); Zhang et al. (2025).
- `ARC`: The adaptive cubic regularized Newton method, originally proposed by Nesterov and Polyak (2006) and Cartis et al. (2011a). A scalable version is available in Dussault et al. (2024), implemented as the open-source package `AdaptiveRegularization.jl`. We adopt its default parameter settings, including its built-in subproblem solver.

All methods use exact Hessian evaluations to ensure fair and consistent comparison.

6.1 Convex optimization

We conduct experiments on the regularized logistic regression problem, which is defined as follows.

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} \log\left(1 + \exp\left(-b_i a_i^\top x\right)\right) + \frac{\gamma}{2}\|x\|^2, \tag{30}$$

where $a_i \in \mathbb{R}^n$ and $b_i \in \{-1, 1\}$, $\gamma = 10^{-5}$ is the regularization parameter. We report the number of Hessian evaluations required by each method on several LIBSVM datasets.[3]

From the results, `HAR-C`, `HAR-S`, and `ARC` generally outperform the other methods across different instances, and overall, their performances are comparable. Interestingly, the budget size $\mathcal{B}$ plays a crucial role in practical performance. In general, a larger $\mathcal{B}$ yields more stable results, whereas, as observed in all six instances, a smaller $\mathcal{B}$ appears to favor the sliding variant (`HAR-S`) significantly.

6.2 CUTEst dataset

We evaluate our methods against `ARC` on the CUTEst dataset; see Gould et al. (2015). For simplicity, we restrict attention to problems with dimension $n \leq 200$, yielding a total of 95 instances.

Table 1 summarizes the tested algorithms. As before, the number in the bracket means the budget size used in the method. Based on the knowledge we gain from convex optimization, we

[3]For details, see `https://www.csie.ntu.edu.tw/cjlin/libsvmtools/datasets/`.

set the budget size to $\mathcal{B} = 15$ to `HAR-C`, while $\mathcal{B} = 5$ to `HAR-S`. Here, K denotes the number of successful instances. We report performance statistics using scaled geometric means (SGMs): $\bar{t}_G$, $\bar{k}_G^f$, $\bar{k}_G^g$, and $\bar{k}_G^H$ represent the (geometric) mean running time, and the number of function, gradient, and Hessian evaluations, respectively.

Table 1: Performance of different algorithms on the CUTEst dataset. $\bar{t}_G$ is scaled by 1 second, and $\bar{k}_G^f$, $\bar{k}_G^g$, and $\bar{k}_G^H$ are scaled by 50 iterations. For failed instances, the iteration count and solving time are set to 20,000.

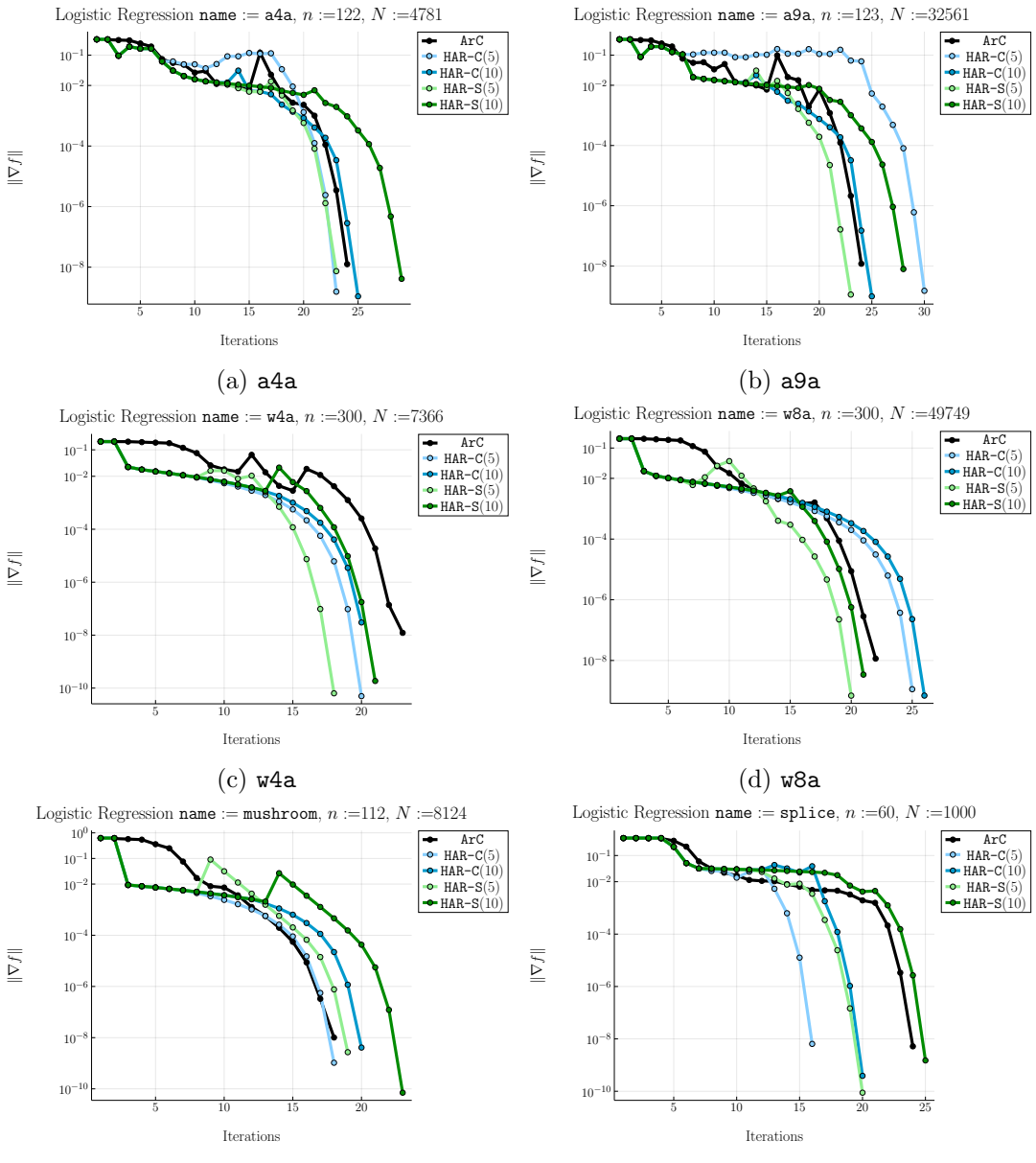
Method	K	$\bar{t}_G$	$\bar{k}_G^f$	$\bar{k}_G^g$	$\bar{k}_G^H$
`ARC`	92	0.26	47.86	46.11	44.85
`HAR-C` (15)	91	0.59	57.34	58.73	55.95
`HAR-S` (5)	90	0.46	48.38	49.70	47.05

From these results, we observe that `HAR-C` and `HAR-S` are broadly comparable to the state-of-the-art `ARC` implementation. Although they solve one and two fewer instances, respectively, their geometric mean statistics are quite similar, aside from the higher running time. This slowdown primarily stems from our current cubic subproblem solver and remains an avenue for optimization.

Beyond SGM-based metrics, we also employ performance profiles on gradient and Hessian evaluations (Dolan et al., 2006); see Figure 2. At point α, the performance profile represents the fraction of problems solved within 2^α times the best iteration count among all methods. These preliminary results suggest that both `HAR-C` and `HAR-S`, especially the latter, are highly competitive. They motivate further exploration of practical improvements to `HAR`, such as adopting a dynamic budget size $\mathcal{B}$.

7 Conclusion

In this work, we develop a new adaptive regularization strategy called the history-aware adaptive regularization method, which uses a history of local Lipschitz estimates to select the current regularization parameter. Our approach works for both convex and nonconvex objectives, with the same complexity guarantees as the standard pth-order tensor method that requires a known Lipschitz constant. Additionally, an Nesterov's accelerated version is proposed for convex optimization. For practical considerations, we introduce two variants that use only a portion of the past Lipschitz estimates. Given a rough upper bound on the Lipschitz constant, these two variants achieve the same iteration complexity guarantees in terms of input accuracy as the original one. In numerical experiments, these two variants demonstrate encouraging numerical performance, which validates the effectiveness of our overall approach. Future work includes investigating whether this strategy works for Hölder smoothness, and how to design an optimal version that matches the lower bound in the convex setting.

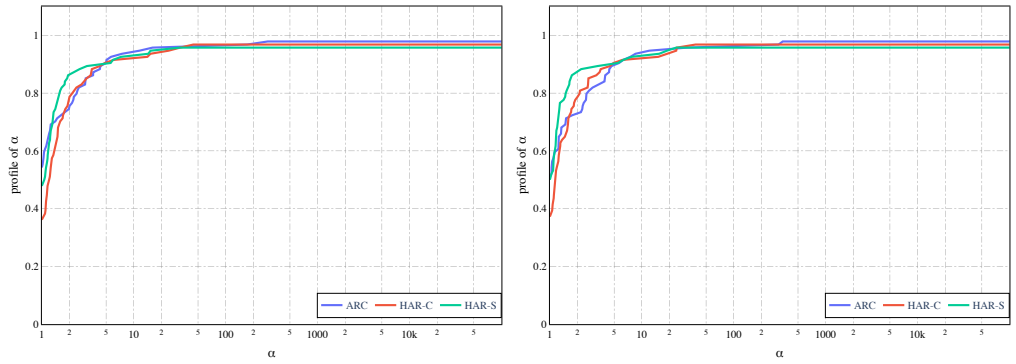


(a) a4a (b) a9a

(c) w4a (d) w8a

(e) mushroom (f) splice

Figure 1: Logistic regression using the LIBSVM datasets. The number in the bracket means the budget size $\mathcal{B}$.



(a) Gradient evaluations (b) Hessian evaluations

Figure 2: Performance profiles of selected CUTEst instances. Panel (a) shows gradient evaluations; panel (b) shows Hessian evaluations.

References

L. Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of mathematics*, 16(1):1–3, 1966. 1

J.-F. Aujol, J. Bigot, and C. Castera. Stochastic adaptive gradient descent without descent. *arXiv preprint arXiv:2509.14969*, 2025. 4

J. Barzilai and J. M. Borwein. Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1):141–148, 1988. 4

A. Beck. *First-order methods in optimization*. SIAM, 2017. 3

E. G. Birgin, J. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163(1):359–368, 2017. 2, 4, 5

Y. Carmon, D. Hausler, A. Jambulapati, Y. Jin, and A. Sidford. Optimal and adaptive monteiro-svaiter acceleration. *Advances in Neural Information Processing Systems*, 35:20338–20350, 2022. 4, 21, 22

C. Cartis and W. Zhu. Cubic-quartic regularization models for solving polynomial subproblems in third-order tensor methods. *Mathematical Programming*, pages 1–53, 2025a. 4

C. Cartis and W. Zhu. Second-order methods for quartically-regularised cubic polynomials, with applications to high-order tensor methods. *Mathematical Programming*, pages 1–47, 2025b. 4

C. Cartis, N. I. Gould, and P. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011a. 3, 22

C. Cartis, N. I. Gould, and P. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. part ii: worst-case function-and derivative-evaluation complexity. *Mathematical programming*, 130(2):295–319, 2011b. 3

C. Cartis, N. I. Gould, and P. L. Toint. Worst-case evaluation complexity and optimality of second-order methods for nonconvex smooth optimization. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3711–3750. World Scientific, 2018. 2, 4, 5

C. Cartis, N. I. Gould, and P. L. Toint. *Evaluation Complexity of Algorithms for Nonconvex Optimization: Theory, Computation and Perspectives*. SIAM, 2022. 2

C. Cartis, R. Hauser, Y. Liu, K. Welzel, and W. Zhu. Efficient implementation of third-order tensor methods with adaptive regularization for unconstrained optimization. *arXiv preprint arXiv:2501.00404*, 2024. 4

X. Chen, B. Jiang, T. Lin, and S. Zhang. Accelerating adaptive cubic regularization of Newton's method via random sampling. *Journal of Machine Learning Research*, 23(90):1–38, 2022. 4, 22

A. R. Conn, N. I. Gould, and P. L. Toint. *Trust region methods*. SIAM, 2000. 2

F. E. Curtis, D. P. Robinson, and M. Samadi. A trust region algorithm with a worst-case iteration complexity of $\mathcal{O}(\epsilon^{-3/2})$ for nonconvex optimization. *Mathematical Programming*, 162(1):1–32, 2017. 3

F. E. Curtis, D. P. Robinson, C. W. Royer, and S. J. Wright. Trust-region Newton-CG with strong second-order complexity guarantees for nonconvex optimization. *SIAM Journal on Optimization*, 31(1):518–544, 2021. 3, 22

N. Doikov and Y. Nesterov. Gradient regularization of Newton method with Bregman distances. *Mathematical programming*, 204(1):1–25, 2024. 3

N. Doikov, K. Mishchenko, and Y. Nesterov. Super-universal regularized Newton method. *SIAM Journal on Optimization*, 34(1):27–56, 2024. 3

E. D. Dolan, J. J. Moré, and T. S. Munson. Optimality measures for performance profiles. *SIAM Journal on Optimization*, 16(3):891–909, Jan. 2006. URL https://epubs.siam.org/doi/10.1137/040608015. 23

J.-P. Dussault, T. Migot, and D. Orban. Scalable adaptive cubic regularization methods. *Mathematical Programming*, 207(1):191–225, Sept. 2024. URL https://doi.org/10.1007/s10107-023-02007-6. 3, 22

W. Gao, Y.-C. Chu, Y. Ye, and M. Udell. Gradient methods with online scaling. *arXiv preprint arXiv:2411.01803*, 2024. 4

A. Garg, R. Kothari, P. Netrapalli, and S. Sherif. Near-optimal lower bounds for convex optimization for all orders of smoothness. *Advances in Neural Information Processing Systems*, 34:29874–29884, 2021. 21

A. Gasnikov, P. Dvurechensky, E. Gorbunov, E. Vorontsova, D. Selikhanovych, C. A. Uribe, B. Jiang, H. Wang, S. Zhang, S. Bubeck, et al. Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In *Conference on Learning Theory*, pages 1392–1393. PMLR, 2019. 4

N. I. M. Gould, D. Orban, and P. L. Toint. Cutest: A constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications*, 60(3):545–557, Apr. 2015. URL https://doi.org/10.1007/s10589-014-9687-3. 22

G. N. Grapiglia and Y. Nesterov. Regularized Newton methods for minimizing functions with Höder continuous Hessians. *SIAM Journal on Optimization*, 27(1):478–506, 2017. 2

G. N. Grapiglia and Y. Nesterov. Tensor methods for minimizing convex functions with Hölder continuous higher-order derivatives. *SIAM Journal on Optimization*, 30(4):2750–2779, 2020. 2

G. N. Grapiglia and Y. Nesterov. Tensor methods for finding approximate stationary points of convex functions. *Optimization Methods and Software*, 37(2):605–638, 2022. 2

A. Griewank. The modification of newton's method for unconstrained optimization by bounding cubic terms. Technical report, Technical report NA/12, 1981. 2

F. Hanzely, N. Doikov, Y. Nesterov, and P. Richtarik. Stochastic subspace cubic newton method. In *International Conference on Machine Learning*, pages 4027–4038. PMLR, 2020. 4

C. He, H. Huang, and Z. Lu. Newton-CG methods for nonconvex unconstrained optimization with Hölder continuous hessian. *Mathematics of Operations Research*, 2025a. 4

C. He, Y. Jiang, C. Zhang, D. Ge, B. Jiang, and Y. Ye. Homogeneous second-order descent framework: a fast alternative to newton-type methods. *Mathematical Programming*, pages 1–62, 2025b. 3

Z. Huang, B. Jiang, and Y. Jiang. Inexact and implementable accelerated newton proximal extragradient method for convex optimization. *arXiv preprint arXiv:2402.11951*, 2024. 4, 22

Y. Ji and G. Lan. High-order accumulative regularization for gradient minimization in convex programming. *arXiv preprint arXiv:2511.03723*, 2025. 21

B. Jiang, T. Lin, and S. Zhang. A unified adaptive tensor approximation scheme to accelerate composite convex optimization. *SIAM Journal on Optimization*, 30(4):2897–2926, 2020. 4, 22

Y. Jiang, C. He, C. Zhang, D. Ge, B. Jiang, and Y. Ye. Beyond nonconvexity: A universal trust-region method with new analyses. *arXiv e-prints*, pages arXiv–2311, 2023. 3

Y. Jiang, C. Zhang, B. Jiang, and Y. Ye. Accelerating trust-region methods: an attempt to balance global and local efficiency, Nov. 2025. URL http://arxiv.org/abs/2511.00680. 3

D. Kovalev and A. Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. *Advances in Neural Information Processing Systems*, 35:35339–35351, 2022. 4, 21

G. Lan, T. Li, and Y. Xu. Projected gradient methods for nonconvex and stochastic optimization: new complexities and auto-conditioned stepsizes. *arXiv preprint arXiv:2412.14291*, 2024. 3, 4

P. Latafat, A. Themelis, L. Stella, and P. Patrinos. Adaptive proximal algorithms for convex optimization under local lipschitz continuity of the gradient. *Mathematical Programming*, pages 1–39, 2024. 4

T. Li and G. Lan. A simple uniformly optimal method without line search for convex optimization: T. li, g. lan. *Mathematical Programming*, pages 1–38, 2025. 2, 3, 4

A. Lucchi and J. Kohler. A sub-sampled tensor method for nonconvex optimization. *IMA Journal of Numerical Analysis*, 43(5):2856–2891, 2023. 4

Y. Malitsky and K. Mishchenko. Adaptive gradient descent without descent. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6702–6712, 2020. 2, 4

Y. Malitsky and K. Mishchenko. Adaptive proximal gradient method for convex optimization. *Advances in Neural Information Processing Systems*, 37:100670–100697, 2024. 4

K. Mishchenko. Regularized Newton method with global $\mathcal{O}(1/k^2)$ convergence. *SIAM Journal on Optimization*, 33(3):1440–1462, 2023. 3

R. D. Monteiro and B. F. Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23 (2):1092–1125, 2013. 4, 21

Y. Nesterov. A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. In *Dokl akad nauk Sssr*, volume 269, page 543, 1983. 2

Y. Nesterov. Accelerating the cubic regularization of newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008. 17

Y. Nesterov. *Lectures on Convex Optimization*, volume 137. Springer, 2018. 5

Y. Nesterov. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, 186(1):157–183, 2021. 2, 4, 10, 17, 18, 19

Y. Nesterov and B. T. Polyak. Cubic regularization of Newton method and its global performance. *Mathematical programming*, 108(1):177–205, 2006. 2, 3, 22

K. Oikonomidis, E. Laude, P. Latafat, A. Themelis, and P. Patrinos. Adaptive proximal gradient methods are universal without approximation. In *International Conference on Machine Learning*, pages 38663–38682. PMLR, 2024. 4

C. W. Royer, M. O'Neill, and S. J. Wright. A newton-cg algorithm with complexity guarantees for smooth unconstrained optimization. *Mathematical Programming*, 180(1):451–488, 2020. 4

J. J. Suh and S. Ma. An adaptive and parameter-free nesterov's accelerated gradient method for convex optimization. *arXiv preprint arXiv:2505.11670*, 2025. 4

M. Weiser, P. Deuflhard, and B. Erdmann. Affine conjugate adaptive newton methods for nonlinear elastomechanics. *Optimisation Methods and Software*, 22(3):413–431, 2007. 2

P. Xu, F. Roosta, and M. W. Mahoney. Newton-type methods for non-convex optimization under inexact hessian information. *Mathematical Programming*, 184(1):35–70, 2020. 4

S. Yagishita and M. Ito. Simple linesearch-free first-order methods for nonconvex optimization. *arXiv preprint arXiv:2509.14670*, 2025. 3, 4, 7

Z. Yao, P. Xu, F. Roosta, S. J. Wright, and M. W. Mahoney. Inexact Newton-cg algorithms with complexity guarantees. *IMA Journal of Numerical Analysis*, 43(3):1855–1897, 2023. 4

Z. Ye, S. Ma, J. Yang, and D. Zhou. A simple adaptive proximal gradient method for nonconvex optimization. *arXiv preprint arXiv:2510.06079*, 2025. 4

C. Zhang, C. He, Y. Jiang, C. Xue, B. Jiang, D. Ge, and Y. Ye. A homogeneous second-order descent method for nonconvex optimization. *Mathematics of Operations Research*, 2025. 3, 22

D. Zhou, S. Ma, and J. Yang. Adabb: Adaptive Barzilai-Borwein method for convex optimization. *Mathematics of Operations Research*, 2025a. 4

J. Zhou, X. Liu, J. Nie, and X. Tang. A tight sdp relaxation for the cubic-quartic regularization problem. *arXiv preprint arXiv:2511.00168*, 2025b. 4

W. Zhu and C. Cartis. Global optimality characterizations and algorithms for minimizing quartically-regularized third-order taylor polynomials. *arXiv preprint arXiv:2504.20259*, 2025. 4