Adaptive Conditional Gradient Sliding: Projection-Free and Line-Search-Free Acceleration

Shota Takahashi
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, Japan
shota@mist.i.u-tokyo.ac.jp

Abstract

We study convex optimization problems over a compact convex set where projections are expensive but a linear minimization oracle (LMO) is available. We propose the *adaptive conditional gradient sliding method* (AdCGS), a projection-free and line-search-free method that retains Nesterov's acceleration with adaptive stepsizes based on local Lipschitz estimates. An optional initialization uses a line search only at the first iteration. AdCGS combines an accelerated outer scheme with an LMO-based inner routine. It reuses gradients across multiple LMO calls to reduce gradient evaluations, while controlling the subproblem inexactness via a prescribed accuracy level coupled with adaptive stepsizes. For convex objective functions, we prove an $\mathcal{O}(\hat{L}_k/k^2)$ bound, yielding accelerated $\mathcal{O}(1/k^2)$ rates when the running local curvature bound $\hat{L}_k$ is bounded, in particular under local smoothness. These rates match projection-based methods without relying on a projection oracle. For locally smooth and locally strongly convex objective functions, we further establish linear convergence without additional geometric assumptions on the constraint set, such as polytopes or strongly convex sets. Experiments on constrained ℓ_p regression, logistic regression, and least-squares problems demonstrate that AdCGS improves over projection-free baselines and provides competitive performance when projections are inexpensive.

1 Introduction

In this paper, we consider constrained convex optimization problems of the form

$$\min_{x \in P} \quad f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable convex function and $P \subset \mathbb{R}^n$ is a compact convex set. Let $x^* \in P$ denote an optimal solution. We assume access to a first-order oracle (FOO) for f, *i.e.*, given $x \in \mathbb{R}^n$, we can compute $\nabla f(x)$, and a linear minimization oracle (LMO) for P, *i.e.*, given $c \in \mathbb{R}^n$, we can compute $v \in \operatorname{argmin}_{u \in P} \langle c, u \rangle$. In large-scale constrained optimization, projection-free methods are particularly useful when projections are computationally expensive or intractable, while LMO calls remain cheap for many structured sets [Combettes and Pokutta, 2021, Braun et al., 2025]. The *conditional gradient method* (CG), also known as the *Frank–Wolfe method* (FW), is widely used for such projection-free settings. However, achieving fast convergence in practice with simple, parameter-free updates remains challenging.

Related work: The CG method was originally proposed by Frank and Wolfe [1956] and independently rediscovered and further developed by Levitin and Polyak [1966]. Early complexity lower bounds for the CG method were established by Canon and Cullum [1968] and refined by GuéLat

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and Marcotte [1986]. More recently, Jaggi [2013] established a refined convergence analysis of CG methods, including a lower bound that highlights a trade-off between sparsity and error. Concurrently, Lan [2013] examined the complexity of linear-optimization-based methods and established a comparable lower bound. Later, Lacoste-Julien and Jaggi [2015] provided a detailed convergence analysis for CG variants.

On the algorithmic side, Lan and Zhou [2016] proposed the *conditional gradient sliding method* (CGS), a projection-free framework that couples an outer accelerated scheme with inner CG loops. A key feature of CGS is that it reduces the number of FOO calls by reusing gradients across multiple LMO calls and thereby skipping some gradient evaluations. A recent improvement of CGS was proposed by Garber [2025]. A line search strategy for CG was introduced by Pedregosa et al. [2020], refined for numerical stability by Pokutta [2024], and extended beyond Lipschitz gradients by Takahashi et al. [2026]. Restart strategies and error-bound analyses have been developed to improve convergence [Kerdreux et al., 2019, 2026, 2022]. See [Pokutta, 2024, Braun et al., 2025] for background on CG methods.

In first-order methods, adaptive stepsizes based on local curvature or Lipschitz estimates have been studied to reduce parameter tuning. More recently, line-search-free methods, *e.g.*, [Malitsky, 2020, Malitsky and Mishchenko, 2020, 2024, Latafat et al., 2025], have emerged as appealing alternatives in practice, using adaptive stepsizes based on local Lipschitz estimates and avoiding parameter tuning. Additionally, CG methods using adaptive stepsizes have been proposed by Yagishita and Ito [2025], Khademi and Silveti-Falls [2025]. The *auto-conditioned fast gradient method* (AC-FGM) [Li and Lan, 2026] is the first method that is line-search-free, combines acceleration with adaptive stepsizes, and provides theoretical guarantees when projections are available. Related accelerated schemes for unconstrained problems have also been studied [Suh and Ma, 2025, Borodich and Kovalev, 2026]. Both CGS and AC-FGM are rooted in Nesterov's accelerated gradient framework [Nesterov, 1983, 2018] and are based on [Auslender and Teboulle, 2006, Tseng, 2008, Gasnikov and Nesterov, 2018, Ahn and Sra, 2022].

Limitations of Existing Methods: Despite significant progress, CGS depends on global smoothness parameters, which are often unknown or difficult to estimate in practice. While AC-FGM is fast and line-search-free, it relies on projections and may be impractical when projections are expensive. Thus, it remains unclear whether accelerated rates can be achieved using adaptive stepsizes without access to a projection oracle. These gaps motivate an accelerated, projection-free, and line-search-free method that explicitly handles the interaction between adaptive stepsizes and inexact inner solves. This raises the following question:

Can we achieve accelerated rates without projection or line-search?

Contributions: We answer the above question in the affirmative and propose the *adaptive conditional gradient sliding method* (AdCGS), an **accelerated**, **projection-free**, and **line-search-free** method. Our main contributions are:

- **Method:** We develop AdCGS, whose inner CG loops use only LMO calls and stop at a prescribed FW-gap tolerance. By employing adaptive stepsizes based on local Lipschitz estimates, AdCGS avoids line search in its main iterations and does not require a global smoothness constant, unlike CGS.
- **Key technical mechanism:** Local Lipschitz estimates allow AdCGS to take larger stepsizes, thereby making progress more sensitive to inner inexactness and typically requiring tighter FW-gap tolerances, hence more LMO calls. Our analysis quantifies this trade-off and provides a termination schedule that preserves acceleration. In particular, for the fixed-iteration scheme, the FW-gap tolerance decays at the same asymptotic rate as in CGS, even with adaptive stepsizes.
- **Theory and convergence rates:** We establish an $\mathcal{O}(\hat{L}_k/k^2)$ bound for convex objective functions and an accelerated $\mathcal{O}(1/k^2)$ rate when $\hat{L}_k$ is bounded, in particular under local smoothness, matching the corresponding projection-based guarantees [Li and Lan, 2026] while using only an LMO and no projection oracle. We also prove linear convergence for locally smooth and locally strongly convex objective functions, without requiring additional structure on the constraint set, such as polytopes or strongly convex sets.

- **Experiments:** Experiments on constrained ℓ_p regression, logistic regression with real-world datasets, and least-squares problems show that AdCGS improves over projection-free baselines and is competitive with projection-based methods.

Notation: We use $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ for the Euclidean norm and inner product, and $\|\cdot\|_p$ for the ℓ_p norm ($p \geq 1$). For symmetric $A \in \mathbb{R}^{n\times n}$, $\lambda_{\max}(A)$ is its largest eigenvalue.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be differentiable and convex. For $x, y \in \mathbb{R}^n$, define $D_f(x, y) := f(x) - f(y) - \langle\nabla f(y), x-y\rangle$. We say f is L-smooth if there exists $L > 0$ such that $\|\nabla f(x)-\nabla f(y)\| \leq L\|x-y\|$ for all $x, y \in \mathbb{R}^n$, which implies $\frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2 \leq D_f(x, y) \leq \frac{L}{2}\|x - y\|^2$ due to the convexity of f. We say f is μ-strongly convex if there exists $\mu > 0$ such that $f - \frac{\mu}{2}\|\cdot\|^2$ is convex. We say f is locally smooth (resp. locally strongly convex) if for every compact set $C \subset \mathbb{R}^n$ there exists $L > 0$ (resp. $\mu > 0$) such that f is L-smooth (resp. μ-strongly convex) on C. We adopt the convention that $\frac{0}{0} = 0$ and $\frac{a}{0} = +\infty$ for all $a > 0$.

2 Adaptive Conditional Gradient Sliding for Convex Objective Functions

In this section, we propose AdCGS for solving (1).

Projection-free accelerated framework: Our proposed method is based on CGS [Lan and Zhou, 2016], a projection-free accelerated framework that couples an accelerated outer scheme with inner CG loops. A key feature of CGS is that it skips some gradient evaluations, allowing multiple LMO calls to reuse a single gradient and thereby reducing the number of FOO calls. We enhance this CGS framework by incorporating adaptive stepsizes driven by local Lipschitz estimates L_k, as in AC-FGM [Li and Lan, 2026]. Concretely, at each outer iteration, AdCGS approximately solves a strongly convex subproblem via inner CG loops:

$$\min_{z\in P}\left\{\langle\nabla f(x_{k-1}), z\rangle + \frac{1}{2\eta_k}\|z - y_{k-1}\|^2\right\}. \tag{2}$$

We run CG on (2) until its FW gap is at most $\delta_k > 0$, *i.e.*, $\max_{v\in P}\langle\nabla f(x_{k-1})+(z_k-y_{k-1})/\eta_k, z_k-v\rangle \leq \delta_k$, and set z_k to the final iterate. The exact minimizer is the projection of $y_{k-1} - \eta_k\nabla f(x_{k-1})$ onto P, but each inner CG step uses only an LMO to minimize a linear model of this quadratic subproblem. A single CG step need not meet the prescribed tolerance; repeated inner steps reuse the same outer gradient, without invoking a projection oracle. We then update $y_k = (1 - \beta_k)y_{k-1} + \beta_k z_k \in P$ and $x_k = \frac{\tau_k}{1+\tau_k}x_{k-1} + \frac{1}{1+\tau_k}z_k \in P$ with $\beta_k \in [0, 1]$ and $\tau_k \geq 0$.

Line-search-free framework: Following AC-FGM [Li and Lan, 2026], we compute $x_1 \neq x_0$ using the initial stepsize η_1, and choose η_k via local Lipschitz estimates:

$$L_1 = \frac{\|\nabla f(x_1) - \nabla f(x_0)\|}{\|x_1 - x_0\|}, \tag{3}$$

$$L_k = \begin{cases} 0, & \text{if } D_f(x_{k-1}, x_k) = 0, \\ \frac{\|\nabla f(x_k)-\nabla f(x_{k-1})\|^2}{2D_f(x_{k-1},x_k)}, & \text{if } D_f(x_{k-1}, x_k) > 0, \end{cases} \quad \text{for } k \geq 2. \tag{4}$$

We present our method in Algorithm 1, which allows arbitrary nonnegative sequences $\{\beta_k\}$, $\{\delta_k\}$, and $\{\tau_k\}$ with $\{\eta_k\}$ based on L_k, defined in Corollaries 3.3, 3.5, and 3.6.

Complexity of inner CG iterations: We analyze the complexity of inner CG loops for (2) to an accuracy level of $\delta_k > 0$. The proof is given in Appendix A.1.

Theorem 2.1. *Let $D := \max_{x,y\in P}\|x - y\|$ be the diameter of P. The number of CG iterations to solve (2) up to accuracy δ_k in the k-th outer iteration is at most $T_k := \left\lceil\frac{6D^2}{\eta_k\delta_k}\right\rceil$.*

3 Convergence Analysis for Convex Objective Functions

In this section, we establish an $\mathcal{O}(\hat{L}_k/k^2)$ bound under Corollary 3.3, which yields $\mathcal{O}(1/k^2)$ when $\hat{L}_k$ is bounded, in particular under local smoothness. Building on a one-step inequality, the main new

Algorithm 1 Adaptive Conditional Gradient Sliding method (AdCGS)

Input: $x_0 = y_0 = z_0 \in P$; $\{\beta_k\}$, $\{\delta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$.
for $k = 1, \ldots,$ **do**
  $z_k = \mathrm{CG}(\nabla f(x_{k-1}), y_{k-1}, \eta_k, \delta_k)$
  $y_k = (1-\beta_k)y_{k-1} + \beta_k z_k$
  $x_k = \frac{\tau_k}{1+\tau_k} x_{k-1} + \frac{1}{1+\tau_k} z_k$
  Compute L_k using (3) for $k = 1$ and (4) for $k \geq 2$.
  Set η_{k+1} according to the stepsize rule based on L_k.
end for
Output: x_k
procedure $\mathrm{CG}(g, u, \eta, \delta)$
  $u_1 = u$
  for $t = 1, \ldots,$ **do**
    $v_t \in \operatorname{argmin}_{v \in P} \langle g + (u_t - u)/\eta, v \rangle$
    if $\max_{v \in P} \langle g + (u_t - u)/\eta, u_t - v \rangle \leq \delta$ **then**
      return u_t
    end if
    $\gamma_t = \min\left\{1, \frac{\langle g + (u_t - u)/\eta, u_t - v_t \rangle}{\|v_t - u_t\|^2/\eta}\right\}$, $u_{t+1} = (1-\gamma_t)u_t + \gamma_t v_t$
  end for
end procedure

ingredient is inexactness in z_k: the FW-gap tolerance δ_k generates an error term that accumulates over iterations. We show how to bound this accumulated error while still allowing adaptive stepsizes, and this yields our convergence rates. We now state the one-step inequality under the following conditions on the parameter sequences $\{\tau_k\}$, $\{\beta_k\}$, and $\{\eta_k\}$: $\tau_1 = 0$,

$$\beta_1 = 0, \quad \beta_k = \beta > 0, k \geq 2, \tag{5}$$

$$\eta_2 \leq \min\left\{(1-\beta)\eta_1, \tfrac{1}{4L_1}\right\}, \tag{6}$$

and, for $k \geq 3$,

$$\eta_k \leq \min\left\{2(1-\beta)^2\eta_{k-1}, \tfrac{\tau_{k-1}}{4L_{k-1}}, \tfrac{\tau_{k-2}+1}{\tau_{k-1}}\eta_{k-1}\right\}, \tag{7}$$

where L_k is the local Lipschitz estimate in (3) and (4).

Proposition 3.1. *Suppose that L_k is defined in (3) and (4), and that $\{\tau_k\}$, $\{\beta_k\}$, and $\{\eta_k\}$ satisfy $\tau_1 = 0$, (5), (6), and (7). Let $\{x_k\}$, $\{y_k\}$, and $\{z_k\}$ be the sequences generated by AdCGS. Then, for any $z \in P$, it holds that*

$$\sum_{i=1}^{k} \eta_{i+1}(\tau_i f(x_i) + \langle \nabla f(x_i), x_i - z\rangle - \tau_i f(x_{i-1})) + \frac{\eta_2}{2\eta_1}(\|z_1 - y_1\|^2 + \|z_2 - z_1\|^2) - \mathcal{S}_k$$
$$\leq \frac{1}{2\beta}\|y_1 - z\|^2 - \frac{1}{2\beta}\|y_{k+1} - z\|^2 + \sum_{i=2}^{k+1} \mathcal{D}_i, \tag{8}$$

where $\mathcal{D}_i := \eta_i\langle \nabla f(x_{i-1}) - \nabla f(x_{i-2}), z_{i-1} - z_i\rangle - \frac{\eta_i\tau_{i-1}}{2L_{i-1}}\|\nabla f(x_{i-1}) - \nabla f(x_{i-2})\|^2 - \frac{1}{2}\|z_i - y_{i-1}\|^2$ for any $i \geq 2$ and $\mathcal{S}_k := \sum_{i=1}^{k} \eta_{i+1}(\delta_{i+1} + \delta_i)$.

Sketch of Proof. The inner loop yields the FW gap of (2) up to δ_k, namely, it satisfies $\langle \eta_k \nabla f(x_{k-1}) + (z_k - y_{k-1}), z - z_k\rangle \geq -\eta_k\delta_k$ for all $z \in P$ and $k \geq 1$. Substituting this inequality into the accelerated scheme and telescoping as in [Li and Lan, 2026, Proposition 1] yields (8). A complete proof is given in Appendix A.2. □

The right-hand side of (8) contains $\sum_{i=2}^{k+1} \mathcal{D}_i$, which must be bounded to obtain convergence rates. Using the same bounding technique as in [Li and Lan, 2026, Theorem 1], we obtain the following theorem. The proof is given in Appendix A.3.

Theorem 3.2. *Suppose that L_k is defined in (3) and (4), and that $\{\tau_k\}$, $\{\beta_k\}$, and $\{\eta_k\}$ satisfy $\tau_1 = 0$, (5), (6), and (7). Then, the sequences $\{x_k\}$, $\{y_k\}$, and $\{z_k\}$ generated by AdCGS are bounded. Moreover, it holds that*

$$f(x_k) - f(x^*) \le \frac{1}{(\tau_k+1)\eta_{k+1}}(\mathcal{E}+\mathcal{S}_k) \quad \text{and} \quad f(\overline{x}_k) - f(x^*) \le \frac{1}{\sum_{i=2}^{k+1}\eta_i}(\mathcal{E}+\mathcal{S}_k),$$

where $\mathcal{E} := \frac{1}{2\beta}\|z_0 - x^*\|^2 + \frac{\eta_2}{2}(5L_1/2 - 1/\eta_1)\|z_1 - z_0\|^2$ and $\overline{x}_k := (\sum_{i=1}^{k-1}((\tau_i+1)\eta_{i+1} - \tau_{i+1}\eta_{i+2})x_i + (\tau_k+1)\eta_{k+1}x_k)/(\sum_{i=2}^{k+1}\eta_i)$, with $\mathcal{S}_k$ defined in Proposition 3.1.

Using Theorem 3.2, we now establish convergence rates of AdCGS for convex objective functions under adaptive stepsizes and inexact inner solves controlled by $\{\delta_k\}$. Here, $\mathcal{S}_k$ collects the error contributions induced by the inner tolerances $\{\delta_k\}$, and a key step in the next result is to upper bound this accumulation explicitly. Moreover, δ_k determines the inner computational effort needed to obtain z_k: by Theorem 2.1, the k-th subproblem can be solved with at most $T_k = \lceil 6D^2/(\eta_k\delta_k)\rceil$ iterations of the inner CG routine, *i.e.*, LMO calls. The corollaries below provide simple schedules for $\{\delta_k\}$ that keep $\mathcal{S}_k$ under control while preserving acceleration. The proof is given in Appendix A.4.

Corollary 3.3. *Suppose that L_k is defined in (3) and (4). Let $\tau_1 = 0$ and $\tau_k = k/2$ for all $k \ge 2$, let β_k be given by (5) with $\beta \in (0, 1-\sqrt{6}/3]$, and let $\delta_k = \frac{D_0^2}{k^{1+\theta}(k+1)}$, where $D_0 := \|z_0 - x^*\|$ and $\theta > 0$. Let $\{\eta_k\}$ satisfy*

$$\eta_k = \begin{cases} \min\{(1-\beta)\eta_1, \frac{1}{4L_1}\}, & \text{if } k = 2, \\ \min\{\eta_2, \frac{1}{4L_2}\}, & \text{if } k = 3, \\ \min\left\{\frac{k}{k-1}\eta_{k-1}, \frac{k-1}{8L_{k-1}}\right\}, & \text{if } k \ge 4. \end{cases}$$

Let $\{x_k\}$, $\{y_k\}$, and $\{z_k\}$ be the sequences generated by AdCGS. Then, it holds that $\eta_k \ge \frac{k}{12\hat{L}_{k-1}}$ for all $k \ge 2$, where $\hat{L}_k := \max\left\{\frac{1}{4(1-\beta)\eta_1}, \max_{1\le i\le k} L_i\right\}$,

$$f(x_k) - f(x^*) \le \frac{12\hat{L}_k}{k(k+1)}\mathcal{R}_1, \quad \text{and} \quad f(\overline{x}_k) - f(x^*) \le \frac{1}{\sum_{i=2}^{k+1}\frac{i}{6\hat{L}_{i-1}}}\mathcal{R}_1,$$

where $\mathcal{R}_1 := 2\,\mathcal{E} + \frac{(1-\beta)(2+\theta)\eta_1 D_0^2}{\theta}$, with $\mathcal{E}$ and $\overline{x}_k$ defined in Theorem 3.2.

Remark 3.4 (On the choice of δ_k). The schedule of δ_k in Corollary 3.3 uses D_0, which is unknown in practice. In the analysis, D_0 can be replaced by the diameter D of P, and the same argument only requires controlling S_k. Moreover, we define $\tilde{L}_k := \max_{2\le i\le k} L_i$ if $\max_{2\le i\le k} L_i > 0$ and $\tilde{L}_k := L_1$ otherwise. In our implementations, we use $\tilde{\delta}_k = \frac{\tilde{L}_k D^2}{k^{1+\theta}(k+1)}$ or $\delta_{1:k} = \frac{L_{1:k}D^2}{k^{1+\theta}(k+1)}$, where $L_{1:k}$ denotes the most recent nonzero value among $L_1, \dots, L_k$. If $\tilde{L}_k$ is bounded, these choices preserve the convergence bounds up to constant factors. Since $\delta_{1:k} \le \tilde{\delta}_k$, $\delta_{1:k}$ generally requires more LMO calls.

Formally setting $\theta = 0$ in Corollary 3.3, we obtain $\mathcal{S}_k \le \eta_2 D_0^2(1 + 2\log(k+1))/2$ and thus $\mathcal{O}(\hat{L}_k \log k/k^2)$, or $\mathcal{O}(\log k/k^2)$ when $\hat{L}_k$ is bounded. Larger θ improves the factor $(2+\theta)/\theta$ in the bound but tightens the inner tolerance, typically increasing the number of LMO calls. A practical range is $\theta \in (0, 1]$; we use $\theta = 10^{-3}$ to prioritize LMO efficiency. When we fix the number of iterations N and use $\delta_k = \frac{D_0^2}{Nk}$, we have the simpler bound. The proof is given in Appendix A.5.

Corollary 3.5. *Suppose that L_k is defined in (3) and (4), and fix the number of outer iterations $N \ge 1$. Let $\delta_k = \frac{D_0^2}{Nk}$ and let $\{\beta_k\}$, $\{\tau_k\}$, and $\{\eta_k\}$ follow the same rule as in Corollary 3.3. Let $\{x_k\}$, $\{y_k\}$, and $\{z_k\}$ be the sequences generated by AdCGS. Then, it holds that*

$$f(x_N) - f(x^*) \le \frac{12\hat{L}_N}{N(N+1)}\left(2\,\mathcal{E} + 3(1-\beta)\eta_1 D_0^2\right),$$

with $\mathcal{E}$ defined in Theorem 3.2. Here D_0 may be any upper bound on $\|z_0 - x^*\|$.

The stepsize rule in Corollary 3.3 is line-search-free. While η_1 is free to choose, overly large (resp., small) values may yield a bound with a $\|z_1 - z_0\|^2$ term in $\mathcal{E}$ (resp., make $((1-\beta)\eta_1)^{-1}$ dominate $\hat{L}_k$), so η_1 should be set at a reasonable scale. We consider two options for initializing η_1.

Algorithm 2 Line Search at the First Iteration for AdCGS

Input: $z_0 \in P$, $\beta \in (0, 1 - \frac{\sqrt{6}}{3}]$, $\delta_1 > 0$, and $L > 0$
Choose $0 < L_0 \leq L$ (*e.g.*, (9)), and $\gamma > 1$
for $j = 0, 1, 2, \ldots$ **do**
  $\eta_{1,j} = \frac{1}{4(1-\beta)L_0\gamma^j}$, $z_{1,j} = \mathrm{CG}(\nabla f(z_0), z_0, \eta_{1,j}, \delta_1)$, $L_{1,j} := \frac{\|\nabla f(z_{1,j}) - \nabla f(z_0)\|}{\|z_{1,j} - z_0\|}$
  if $\eta_{1,j} \leq \frac{2}{5L_{1,j}}$ **then**
    return $z_{1,j}, \eta_{1,j}, L_{1,j}$
  end if
end for

(i) Line-search-free initialization: Set $\eta_1 = \frac{c}{4(1-\beta)L_0}$ for some $c > 0$ and $L_0 > 0$, where L_0 is an initial Lipschitz estimate [Li and Lan, 2026]. For example, take $z_{-1} \in P$ with $z_{-1} \neq z_0$ (a perturbation of z_0) and set

$$L_0 := \frac{\|\nabla f(z_{-1}) - \nabla f(z_0)\|}{\|z_{-1} - z_0\|}. \tag{9}$$

(ii) Line search at the first iteration: To avoid an explicit dependence on the initial displacement (*e.g.*, $\|z_1 - z_0\|^2$), perform a line search only at the first iteration to find η_1 satisfying $\eta_1 \leq \frac{2}{5L_1}$. The procedure is given in Algorithm 2. Assume that f is locally smooth. Since $\{x_k\}$ is bounded by Corollary 3.3, f is L-smooth on the set of iterates. For $\gamma \geq 8/5$, the line search guarantees $\frac{2}{5\gamma L} \leq \eta_1 \leq \frac{2}{5L_1}$, which implies $\hat{L}_k \leq \frac{5\gamma L}{8(1-\beta)}$. Consequently, we obtain

$$f(x_k) - f(x^*) \leq \frac{15\gamma L\sigma D_0^2}{4k(k+1)} \quad \text{and} \quad f(\overline{x}_k) - f(x^*) \leq \frac{15\gamma L\sigma D_0^2}{4k(k+3)},$$

where $\sigma = \frac{2}{\beta(1-\beta)} + \frac{2(2+\theta)\eta_1}{\theta}$. Algorithm 2 requires at most $\mathcal{O}(\log_\gamma(L/L_0))$ iterations.

Li and Lan [2026] proposed an alternative choice of τ_k that allows larger stepsizes. We also derive accelerated rates. The proof is given in Appendix A.6.

Corollary 3.6. *Suppose that L_k is defined in (3) and (4). Let $\tau_1 = 0$, $\tau_2 = 1$, and $\tau_k = \tau_{k-1} + \frac{\alpha}{2} + \frac{2(1-\alpha)\eta_k L_{k-1}}{\tau_{k-1}}$ for all $k \geq 3$ with a constant $\alpha \in [0, 1]$, let β_k be given by (5) with $\beta \in (0, 1 - \sqrt{6}/3]$, and let $\delta_k = \frac{D_0^2}{k^{1+\theta}(k+1)}$ for some $\theta > 0$. Let $\{\eta_k\}$ satisfy*

$$\eta_k = \begin{cases} \min\left\{(1-\beta)\eta_1, \frac{1}{4L_1}\right\}, & \text{if } k = 2, \\ \min\left\{\frac{4}{3}\eta_{k-1}, \frac{\tau_{k-2}+1}{\tau_{k-1}}\eta_{k-1}, \frac{\tau_{k-1}}{4L_{k-1}}\right\}, & \text{if } k \geq 3. \end{cases}$$

Let $\{x_k\}$, $\{y_k\}$, and $\{z_k\}$ be the sequences generated by AdCGS. Then, it holds that

$$f(x_k) - f(x^*) \leq \frac{12\hat{L}_k\mathcal{R}_2}{(\alpha k + 4 - 2\alpha)(\alpha k + 3 - 2\alpha)} \quad \text{and} \quad f(\overline{x}_k) - f(x^*) \leq \frac{12\hat{L}_k\mathcal{R}_2}{6k + \alpha k(k-3)},$$

where $\mathcal{R}_2 := 2\mathcal{E} + \frac{D_0^2(2+\theta)}{4\theta\underline{L}_k}$ and $\underline{L}_k := \min_{1\leq i\leq k} L_i$, with $\mathcal{E}$ and $\overline{x}_k$ as in Theorem 3.2, and $\hat{L}_k$ as in Corollary 3.3.

When $\alpha = 1$, Corollary 3.6 recovers Corollary 3.3. For $\alpha \in (0, 1)$, it gives accelerated variants with different constants, whereas $\alpha = 0$ is included only as an empirical variant and sometimes performs well in practice (see, *e.g.*, Figure 4).

4 Linear Rate for Locally Strongly Convex Objective Functions

In this section, we establish the linear convergence of AdCGS for locally strongly convex objective functions. We work under the assumption that f is locally strongly convex and locally smooth. We present AdCGS for locally strongly convex objective functions in Algorithm 3, which repeatedly

Algorithm 3 AdCGS for locally strongly convex problems

Input: $w_0 \in P$; $\gamma \geq 8/5$; an estimate $\phi_0 > 0$ satisfying $f(w_0) - f(x^*) \leq \phi_0$
for $s = 1, \ldots,$ **do**
  Let w_s be the last point output by the AdCGS(Algorithm 1 with Algorithm 2) when run for

$$N = \left\lceil \sqrt{\frac{30\gamma L}{\mu}\left(\frac{1}{\beta(1-\beta)} + \frac{3}{2}\right)} \right\rceil$$

  iterations with parameters L and μ given by local smoothness and local strong convexity, $x_0 = w_{s-1}$, $\eta_1 > 0$, β_k defined in (5), $\delta_k = \frac{(1-\beta)\phi_0}{2^s \mu N \eta_k}$, η_k and τ_k defined in Corollary 3.3, and L_k defined in (3) and (4). In Algorithm 2, recompute δ_1 using each trial stepsize $\eta_{1,j}$.
end for
Output: w_s

runs Algorithm 1 for a fixed number of iterations N. We then establish the linear convergence rate of Algorithm 3. Its proof can be found in Appendix A.7. The following oracle-complexity bounds exclude rejected initialization trials.

Theorem 4.1. *Suppose that f is locally strongly convex and locally smooth. Let $\{w_s\}$ be the sequence generated by Algorithm 3. Then, $f(w_s) - f(x^*) \leq \frac{\phi_0}{2^s}$ holds for $s \geq 0$. Equivalently, to achieve $f(w_s) - f(x^*) \leq \epsilon$, the total number of FOO calls is at most $O(N_0 \log_2\lceil \max\{\phi_0/\epsilon, 1\}\rceil)$, and the total number of LMO calls is at most $O(LD^2/\epsilon + N_0 \log_2\lceil \max\{\phi_0/\epsilon, 1\}\rceil)$, where $N_0 = \sqrt{\frac{\gamma L}{\mu}\left(\frac{1}{\beta(1-\beta)} + 1\right)}$.*

Similarly, by choosing a different value of N, one can establish linear convergence when τ_k and η_k are defined as in Corollary 3.6. While linear convergence rates are available under additional geometric assumptions, such as polytopes [Lacoste-Julien and Jaggi, 2015], strongly convex sets [Garber and Hazan, 2015], or uniformly convex sets [Kerdreux et al., 2021], our result applies to an arbitrary compact convex set P and highlights an oracle-complexity trade-off. Compared with CG, both CGS and AdCGS substantially reduce the number of FOO calls at the cost of additional LMO calls.

5 Numerical Experiments

In this section, we present numerical experiments to evaluate the performance of our algorithm. All experiments were run in Julia 1.12 using FrankWolfe.jl [Besançon et al., 2022][*] on a Mac Studio equipped with an Apple M4 Max processor and 128GB of LPDDR5 memory. In the plots, AdCGS denotes AdCGS with $\alpha = 1.0$. Labels AdCGS:0.5, AdCGS:0.1, and AdCGS:0.0 correspond to $\alpha = 0.5, 0.1, 0.0$, respectively. We use $\beta = 1 - \sqrt{6}/3$, $\eta_1 = \frac{2}{5L_0}$ with L_0 defined in (9), and $\theta = 10^{-3}$. The effect of choosing θ was small in our experiments, possibly because we use D in place of D_0, which relaxes the inner tolerance. Methods with the suffix "(LS1)" (*e.g.*, AdCGS:0.5(LS1)) apply the line search (Algorithm 2) at the first iteration. All other AdCGS runs are fully line-search-free. We stop when $\max_{v \in P}\langle \nabla f(x_k), x_k - v\rangle \leq 10^{-10}$ or when the maximum number of iterations is reached. Each figure reports both the primal gap and the FW gap, with all y-axes on a logarithmic scale. When $f(x^*)$ is unknown, we use the lowest objective value attained by any method as a proxy for $f(x^*)$ when computing the primal gap. We report results versus the number of iterations (FOO calls), wall-clock time, and the number of LMO calls. For readability, we truncate the y-axis to omit extremely large values. Appendix B.4 presents additional experiments on Poisson regression over positive semidefinite matrices with a trace constraint, and Appendix B.5 reports LMO calls per gradient evaluation.

5.1 Least Squares Regression over the Simplex

We first consider the least squares regression problem over the simplex, $\min_{x \in \Delta_n} \frac{1}{2}\|Ax - b\|^2$, where the simplex $\Delta_n := \{x \in \mathbb{R}^n_+ \mid \sum_{i=1}^n x_i \leq 1\}$, $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$. We generated A with

[*] https://github.com/ZIB-IOL/FrankWolfe.jl

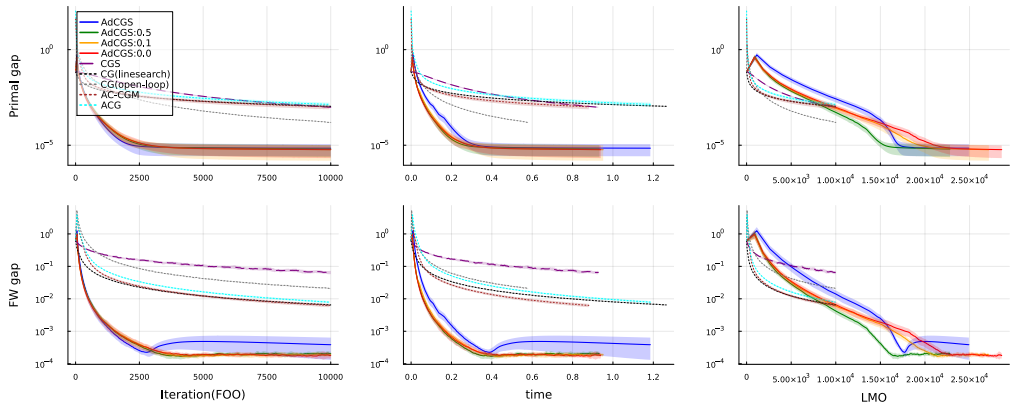


Figure 1: Least squares regression over the simplex with $(m, n) = (2500, 500)$ compared with existing methods.

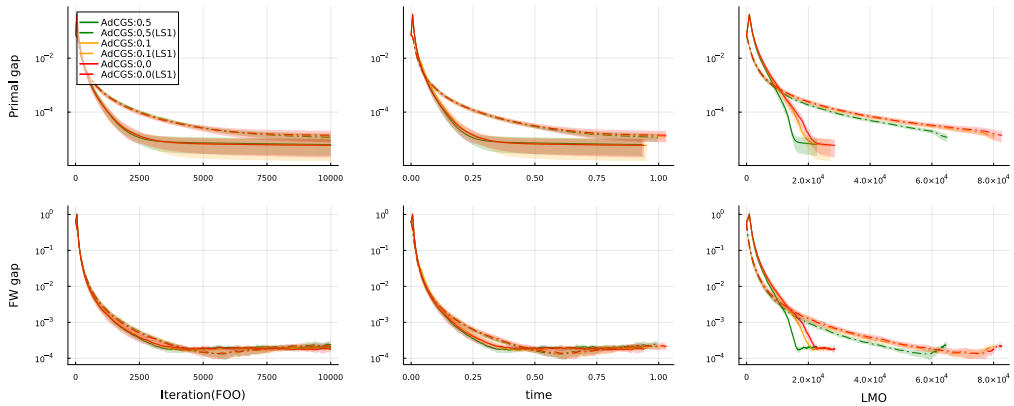


Figure 2: Least squares regression over the simplex with $(m, n) = (2500, 500)$ compared with the AdCGS variants.

i.i.d. entries from the uniform distribution on $[0, 1]$, drew a random vector uniformly from $[0, 1]^n$ and projected it onto Δ_n to obtain x^*, and defined $b := Ax^*$. The function f is L-smooth with $L = \lambda_{\max}(A^\top A)$. We compared AdCGS(Algorithm 1) with CGS [Lan and Zhou, 2016], CG with line search [Pedregosa et al., 2020], CG with open-loop stepsizes (using $2/(2 + k)$, *e.g.*, [Pokutta, 2024, Algorithm 4]), the auto-conditioned CG method (AC-CGM) [Yagishita and Ito, 2025], and the adaptive CG method (ACG) [Khademi and Silveti-Falls, 2025]. ACG also employs Lipschitz estimates but still relies on backtracking at every iteration. AdCGS uses $\tilde{\delta}_k$ as the inner tolerance (see Remark 3.4). We set $x_0 = (\frac{1}{n}, \ldots, \frac{1}{n})$ and the maximum number of iterations to 10000. We consider the cases $(m, n) = (1000, 200)$ and $(m, n) = (2500, 500)$. We conducted experiments on 20 independently generated random instances (random seeds 1–20). In the figures, the solid line shows the mean over the 20 runs, and the shaded area indicates the mean $\pm$ standard deviation. The results for $(m, n) = (2500, 500)$, including the effect of the first-iteration line search (Algorithm 2), are shown in Figures 1 and 2. The remaining results (including those for $(m, n) = (1000, 200)$) are provided in Appendix B.1 (see Figures 5 and 6). AdCGS:0.5 outperformed the other methods and achieved the best performance, while the other AdCGS variants showed similarly strong performance. The first-iteration line search yields comparable performance for all variants, except for AdCGS with $\alpha = 1.0$. AdCGS(LS1) with $\alpha = 1.0$ is omitted because it is unstable. AdCGS requires more LMO calls than CGS because of tighter FW-gap tolerances.

5.2 ℓ_p Loss Regression over the ℓ_2-Ball

We consider the ℓ_p loss regression problem over the ℓ_2-ball, $\min_{x \in P} \|Ax - b\|_p^p$, where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $p > 1$, and $P = \{x \in \mathbb{R}^n \mid \|x\| \leq r\}$. We use the YearPredictionMSD.train

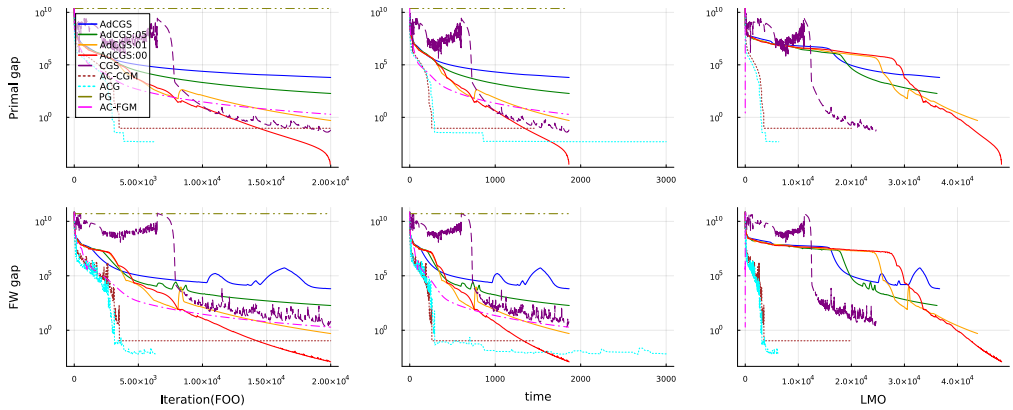


Figure 3: ℓ_p loss regression over the YearPredictionMSD.train dataset with $(m, n) = (463715, 90)$.

($m = 463715$, $n = 90$) and cpusmall ($m = 8192$, $n = 12$) datasets from LIBSVM [Chang and Lin, 2011] to form A and b. We standardize the entries of A. We set $p = 1.5$, $r = 0.97\|\tilde{x}\|$ with $\tilde{x} \in \operatorname{argmin}_{x \in \mathbb{R}^n} \|Ax - b\|$, set x_0 to the zero vector, and set the maximum number of iterations to 20000. We compared AdCGS with CGS, AC-CGM, ACG, the projected gradient method (PG), and AC-FGM ($\alpha = 0.5$). AdCGS uses $\delta_{1:k}$ as the inner tolerance (see Remark 3.4). Standard CG methods became ineffective because the CG updates were nearly zero, so we do not report them in this experiment. We include AC-FGM primarily as a baseline to quantify the performance gap between AC-FGM and AdCGS. When projections are efficiently computable, AC-FGM typically achieves better performance than AdCGS. Results for YearPredictionMSD.train are shown in Figure 3 and results for cpusmall are provided in Appendix B.2 (see Figure 7). Note that gradient computation is expensive for YearPredictionMSD.train because of its large m. AdCGS:0.0 outperformed existing projection-free methods, PG, and AC-FGM. However, AC-FGM ($\alpha = 0.5$) outperformed AdCGS:0.5, likely because AC-FGM computes projections exactly. For $1 < p < 2$, the objective is not locally smooth, so the theoretical $\mathcal{O}(\hat{L}_k/k^2)$ bound need not yield $\mathcal{O}(1/k^2)$; we report empirical results using our curvature-scaled tolerance in this regime. CGS and PG fail to reduce the primal gap in this setting.

5.3 Logistic Regression

We consider the logistic regression problem over the K-sparse polytope, $\min_{x \in P} \sum_{i=1}^{m} \log(1 + \exp(-b_i a_i^\top x))$, where $a_i \in \mathbb{R}^n$ and $b_i \in \{-1, 1\}$ for $i = 1, \ldots, m$ and $P = \{x \in \mathbb{R}^n \mid \|x\|_1 \leq \kappa K, \|x\|_\infty \leq \kappa\}$ with $\kappa > 0$. We use the a9a ($m = 32561$, $n = 123$, $\kappa = 1$), gisette ($m = 6000$, $n = 5000$, $\kappa = 5$), phishing ($m = 11055$, $n = 68$, $\kappa = 5$), and w8a ($m = 49749$, $n = 300$, $\kappa = 1$) datasets from LIBSVM [Chang and Lin, 2011]. We set $K = 0.05n$ and the maximum number of iterations to 20000, and initialized x_0 as the zero vector. We compared AdCGS with CGS, AC-CGM, and ACG. Since the CG updates were negligible, we omit them. AdCGS uses $\delta_{1:k}$ as the inner tolerance (see Remark 3.4). In this experiment, we set the maximum number of inner iterations for AdCGS to 50. The results for gisette are shown in Figure 4. Results for the remaining datasets are provided in Appendix B.3 (see Figures 8-10). AdCGS:0.0 outperformed existing methods.

6 Conclusion

We proposed AdCGS, an accelerated, projection-free, and line-search-free framework with adaptive stepsizes. Our analysis establishes an $\mathcal{O}(\hat{L}_k/k^2)$ bound for convex objective functions, an $\mathcal{O}(1/k^2)$ rate when $\hat{L}_k$ is bounded, and linear convergence for locally smooth and locally strongly convex objective functions. It also clarifies how adaptive stepsizes interact with inexact LMO-based inner solves, yielding oracle-complexity bounds that trade off gradient and LMO calls. Experiments show that AdCGS improves over projection-free baselines and is competitive with projection-based methods when projections are inexpensive.

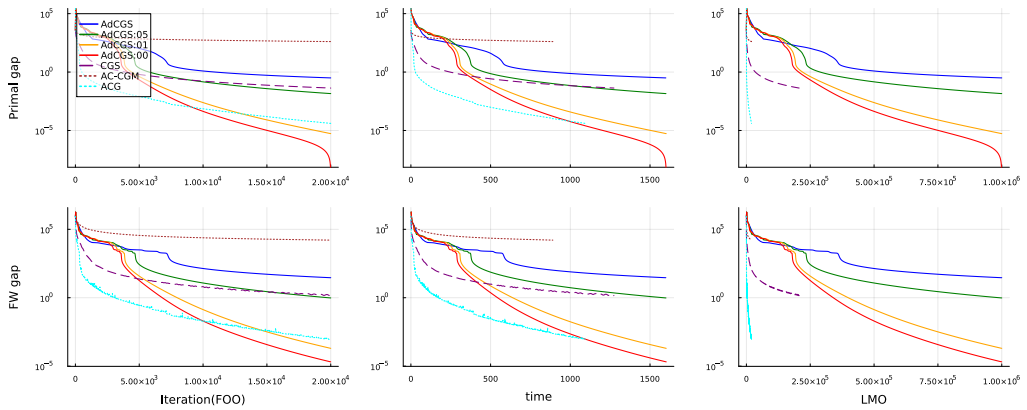


Figure 4: Logistic regression over the gisette dataset with $(m, n) = (6000, 5000)$.

Limitations and future work: Linear convergence generally requires local strong convexity. However, algorithms for estimating the associated local strong convexity parameters remain relatively scarce. Although such parameters could plausibly be estimated in a manner analogous to Lipschitz estimates, a rigorous convergence analysis of these estimation procedures has not yet been established for AdCGS. Developing such estimation procedures remains an important direction for future work. The present analysis covers only two classes of functions: convex functions and locally strongly convex functions. Future work will extend the framework to nonconvex settings, including weakly convex objective functions. Other directions include approximate LMOs, stochastic gradients, and inexact gradient evaluations.

Acknowledgments and Disclosure of Funding

The author is grateful to Akiko Takeda for helpful discussions and constructive feedback. This project has been funded by the Japan Society for the Promotion of Science (JSPS) and the Nakajima foundation; JSPS KAKENHI Grant Number JP25K21156 and JP23K26336. The author declares no competing interests.

References

K. Ahn and S. Sra. Understanding Nesterov's acceleration via proximal point method. In K. Bringmann and T. Chan, editors, *Proc. Symp. Simplicity in Algorithms*, pages 117–130, Philadelphia, PA, 2022. SIAM.

A. Auslender and M. Teboulle. Interior gradient and proximal methods for convex and conic optimization. *SIAM J. Optim.*, 16(3):697–725, 2006.

M. Besançon, A. Carderera, and S. Pokutta. FrankWolfe.jl: A high-performance and flexible toolbox for Frank–Wolfe algorithms and conditional gradients. *INFORMS J. Comput.*, 34(5):2611–2620, 2022.

E. Borodich and D. Kovalev. Nesterov finds GRAAL: Optimal and adaptive gradient method for convex optimization. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust, editors, *Proc. 14th Int. Conf. Learn. Represent.*, pages 95533–95558, Rio de Janeiro, Brazil, 2026.

G. Braun, A. Carderera, C. W. Combettes, H. Hassani, A. Karbasi, A. Mokhtari, and S. Pokutta. *Conditional Gradient Methods: From Core Principles to AI Applications*, volume 35 of *MOS-SIAM Series on Optimization*. SIAM, Philadelphia, PA, 2025.

M. D. Canon and C. D. Cullum. A tight upper bound on the rate of convergence of Frank–Wolfe algorithm. *SIAM J. Control*, 6(4):509–516, 1968.

C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, 2011. Software available at https://www.csie.ntu.edu.tw/~cjlin/libsvm/.

C. W. Combettes and S. Pokutta. Complexity of linear minimization and projection on some sets. *Oper. Res. Lett.*, 49(4):565–571, 2021.

M. Frank and P. Wolfe. An algorithm for quadratic programming. *Nav. Res. Logist. Q.*, 3(1–2): 95–110, 1956.

D. Garber. Accelerated Frank–Wolfe algorithms: Complementarity conditions and sparsity. *arXiv preprint arXiv:2511.02821*, 2025.

D. Garber and E. Hazan. Faster rates for the Frank–Wolfe method over strongly-convex sets. In F. Bach and D. Blei, editors, *Proc. 32nd Int. Conf. Mach. Learn.*, volume 37 of *PMLR*, pages 541–549, Lille, France, 2015. PMLR.

A. V. Gasnikov and Y. Nesterov. Universal method for stochastic composite optimization problems. *Comput. Math. Math. Phys.*, 58(1):48–64, 2018.

J. GuéLat and P. Marcotte. Some comments on Wolfe's 'away step'. *Math. Program.*, 35(1):110–119, 1986.

M. Jaggi. Revisiting Frank–Wolfe: Projection-free sparse convex optimization. In S. Dasgupta and D. McAllester, editors, *Proc. 30th Int. Conf. Mach. Learn.*, volume 28 of *PMLR*, pages 427–435, Atlanta, Georgia, USA, 2013. PMLR.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Restarting Frank–Wolfe. In K. Chaudhuri and M. Sugiyama, editors, *Proc. 22nd Int. Conf. Artif. Intell. Stat.*, volume 89 of *PMLR*, pages 1275–1283, Naha, Okinawa, Japan, 2019. PMLR.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Projection-free optimization on uniformly convex sets. In A. Banerjee and K. Fukumizu, editors, *Proc. 24th Int. Conf. Artif. Intell. Stat.*, volume 130 of *PMLR*, pages 19–27, San Diego, California, USA, 2021. PMLR.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Restarting Frank–Wolfe: Faster rates under Hölderian error bounds. *J. Optim. Theory Appl.*, 192(3):799–829, 2022.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Local and global uniform convexity conditions. In S. Pokutta, editor, *Data Science and Optimization*, volume 91 of *Fields Institute Communications*, pages 85–120. Springer, Cham, Switzerland, 2026.

A. Khademi and A. Silveti-Falls. Adaptive conditional gradient descent. *arXiv preprint arXiv:2510.11440*, 2025.

S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank–Wolfe optimization variants. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Adv. Neural Inf. Process. Syst.*, volume 28, pages 496–504, Montreal, Canada, 2015. Curran Associates, Inc.

G. Lan. The complexity of large-scale convex programming under a linear optimization oracle. *arXiv preprint arXiv:1309.5550*, 2013.

G. Lan and Y. Zhou. Conditional gradient sliding for convex optimization. *SIAM J. Optim.*, 26(2): 1379–1409, 2016.

P. Latafat, A. Themelis, L. Stella, and P. Patrinos. Adaptive proximal algorithms for convex optimization under local Lipschitz continuity of the gradient. *Math. Program.*, 213(1–2):433–471, 2025.

E. S. Levitin and B. T. Polyak. Constrained minimization methods. *U.S.S.R. Comput. Math. Math. Phys.*, 6(5):1–50, 1966.

T. Li and G. Lan. A simple uniformly optimal method without line search for convex optimization. *Math. Program.*, 219(1–2):219–256, 2026.

Y. Malitsky. Golden ratio algorithms for variational inequalities. *Math. Program.*, 184(1–2):383–410, 2020.

Y. Malitsky and K. Mishchenko. Adaptive gradient descent without descent. In H. Daumé, III and A. Singh, editors, *Proc. 37th Int. Conf. Mach. Learn.*, volume 119 of *PMLR*, pages 6702–6712. PMLR, 2020.

Y. Malitsky and K. Mishchenko. Adaptive proximal gradient method for convex optimization. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Adv. Neural Inf. Process. Syst.*, volume 37, pages 100670–100697, Vancouver, Canada, 2024. Curran Associates, Inc.

Y. Nesterov. A method of solving a convex programming problem with convergence rate $O\left(\frac{1}{k^2}\right)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983.

Y. Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, Cham, Switzerland, 2nd edition, 2018.

F. Pedregosa, G. Negiar, A. Askari, and M. Jaggi. Linearly convergent Frank–Wolfe with backtracking line-search. In S. Chiappa and R. Calandra, editors, *Proc. 23rd Int. Conf. Artif. Intell. Stat.*, volume 108 of *PMLR*, pages 1–10, Palermo, Italy, 2020. PMLR.

S. Pokutta. The Frank–Wolfe algorithm: A short introduction. *Jahresber. Dtsch. Math.-Ver.*, 126(1): 3–35, 2024.

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.

S. Takahashi, S. Pokutta, and A. Takeda. Fast Frank–Wolfe algorithms with adaptive Bregman stepsize for weakly convex functions. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust, editors, *Proc. 14th Int. Conf. Learn. Represent.*, pages 78614–78651, Rio de Janeiro, Brazil, 2026.

P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Submitted to SIAM J. Optim., 2008.

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

A Proofs of Theoretical Results

A.1 Proof of Theorem 2.1

The following lemma will be useful in the analysis.

Lemma A.1 ([Lan and Zhou, 2016, Lemma 2.1]). *Let $\lambda_t \in (0,1]$, $t = 1, 2, \ldots$ and let*

$$\Lambda_t = \begin{cases} 1, & \text{if } t = 1, \\ (1-\lambda_t)\Lambda_{t-1}, & \text{if } t \geq 2. \end{cases}$$

Suppose that $\Lambda_t > 0$ for all $t \geq 2$ and the sequence $\{\zeta_t\}$ satisfies, for any $t \geq 1$ and a sequence $\{\nu_t\}$,

$$\zeta_t \leq (1-\lambda_t)\zeta_{t-1} + \nu_t.$$

Then, for any $j \in \{1, \ldots, k\}$, it holds that

$$\zeta_k \leq \Lambda_k \left(\frac{1-\lambda_j}{\Lambda_j}\zeta_{j-1} + \sum_{i=j}^{k} \frac{\nu_i}{\Lambda_i} \right). \tag{10}$$

The proof follows the same argument as that of [Lan and Zhou, 2016, Theorem 2.2(c)]. We include it here for completeness and to make the presentation self-contained. Recall the definition of the actual update $u_{t+1} := (1-\gamma_t)u_t + \gamma_t v_t$, where $\gamma_t = \min\left\{1, \frac{\langle \nabla f(x_{k-1}) + (u_t - u)/\eta_k, u_t - v_t\rangle}{\|v_t - u_t\|^2/\eta_k}\right\}$ is the minimizer of

$$\min_{\gamma \in [0,1]} \quad \langle \nabla f(x_{k-1}), (1-\gamma)u_t + \gamma v_t \rangle + \frac{1}{2\eta_k}\|(1-\gamma)u_t + \gamma v_t - y_{k-1}\|^2.$$

See the CG subroutine in Algorithm 1 for details.

Proof of Theorem 2.1. Let us define $\lambda_t := \frac{2}{t}$ and $\Lambda_t := \frac{2}{t(t-1)}$ for $t \geq 2$. Obviously, $\Lambda_{t+1} := (1-\lambda_{t+1})\Lambda_t$ holds for all $t \geq 2$. We also define

$$\psi_k(z) = \langle \nabla f(x_{k-1}), z \rangle + \frac{1}{2\eta_k}\|z - y_{k-1}\|^2.$$

Let $\overline{u}_{t+1} = (1-\lambda_{t+1})u_t + \lambda_{t+1}v_t$. We use $\gamma_t = \min\left\{1, \frac{\langle \nabla f(x_{k-1}) + (u_t - u)/\eta_k, u_t - v_t\rangle}{\|v_t - u_t\|^2/\eta_k}\right\}$, which is the minimizer of $\min_{\gamma\in[0,1]} \psi_k((1-\gamma)u_t + \gamma v_t)$ along the segment between u_t and v_t, *i.e.*, $\psi_k(u_{t+1}) \leq \psi_k(\overline{u}_{t+1})$ holds for $u_{t+1} := (1-\gamma_t)u_t + \gamma_t v_t$. Using the fact that $\psi_k(z)$ has a Lipschitz continuous gradient with Lipschitz constant $1/\eta_k$, we have

$$\begin{aligned}
\psi_k(u_{t+1}) &\leq \psi_k(\overline{u}_{t+1}) \\
&\leq \psi_k(u_t) + \langle \nabla\psi_k(u_t), \overline{u}_{t+1} - u_t\rangle + \frac{1}{2\eta_k}\|\overline{u}_{t+1} - u_t\|^2 \\
&\overset{\text{(a)}}{=} (1-\lambda_{t+1})\psi_k(u_t) + \lambda_{t+1}(\psi_k(u_t) + \langle \nabla\psi_k(u_t), v_t - u_t\rangle) + \frac{\lambda_{t+1}^2}{2\eta_k}\|v_t - u_t\|^2, \quad (11)
\end{aligned}$$

where (a) follows from $\overline{u}_{t+1} - u_t = \lambda_{t+1}(v_t - u_t)$. By the definition of v_t and the convexity of ψ_k, we have $\psi_k(u_t) + \langle \nabla\psi_k(u_t), v_t - u_t\rangle \leq \psi_k(u_t) + \langle \nabla\psi_k(u_t), z - u_t\rangle \leq \psi_k(z)$ for all $z \in P$. Therefore, from (11), we obtain, for all $z \in P$,

$$\psi_k(u_{t+1}) - \psi_k(z) \leq (1-\lambda_{t+1})(\psi_k(u_t) - \psi_k(z)) + \frac{\lambda_{t+1}^2}{2\eta_k}\|v_t - u_t\|^2. \quad (12)$$

Using (10) with indices shifted by one and (12), we have, for all $z \in P$,

$$\begin{aligned}
\psi_k(u_{t+1}) - \psi_k(z) &\leq \Lambda_{t+1}(1-\lambda_2)(\psi_k(u_1) - \psi_k(z)) + \frac{\Lambda_{t+1}}{\eta_k}\sum_{i=1}^{t}\frac{\lambda_{i+1}^2}{2\Lambda_{i+1}}\|v_i - u_i\|^2 \\
&\overset{\text{(b)}}{\leq} \frac{2D^2}{\eta_k(t+1)}, \quad (13)
\end{aligned}$$

where (b) follows from $\lambda_2 = 1$, $\frac{\lambda_{i+1}^2}{2\Lambda_{i+1}} = \frac{i}{i+1} \leq 1$, and $\|v_i - u_i\| \leq D$ for all $i \geq 1$. On the other hand, the FW gap of ψ_k at u is defined as

$$\mathcal{G}_k(u) := \max_{v\in P}\langle \nabla\psi_k(u), u - v\rangle.$$

Using (11), we have for any $i \geq 1$,

$$\lambda_{i+1}\,\mathcal{G}_k(u_i) \leq \psi_k(u_i) - \psi_k(u_{i+1}) + \frac{\lambda_{i+1}^2}{2\eta_k}\|v_i - u_i\|^2 \leq \xi_i - \xi_{i+1} + \frac{\lambda_{i+1}^2}{2\eta_k}\|v_i - u_i\|^2,$$

where $\xi_i := \psi_k(u_i) - \psi_k^*$ for all $i \geq 1$ and $\psi_k^* := \min_{z\in P}\psi_k(z)$. Dividing both sides by Λ_{i+1} and summing from $i = 2$ to t, we have

$$\begin{aligned}
\sum_{i=2}^{t}\frac{\lambda_{i+1}}{\Lambda_{i+1}}\,\mathcal{G}_k(u_i) &\leq -\frac{1}{\Lambda_{t+1}}\xi_{t+1} + \sum_{i=3}^{t}\left(\frac{1}{\Lambda_{i+1}} - \frac{1}{\Lambda_i}\right)\xi_i + 3\xi_2 + \sum_{i=2}^{t}\frac{\lambda_{i+1}^2}{2\eta_k\Lambda_{i+1}}\|v_i - u_i\|^2 \\
&\leq \sum_{i=3}^{t}\left(\frac{1}{\Lambda_{i+1}} - \frac{1}{\Lambda_i}\right)\xi_i + 3\xi_2 + \sum_{i=2}^{t}\frac{\lambda_{i+1}^2}{2\eta_k\Lambda_{i+1}}D^2 \\
&\overset{\text{(c)}}{\leq} \sum_{i=2}^{t} i\xi_i + \xi_2 + \frac{t-1}{\eta_k}D^2,
\end{aligned}$$

where (c) follows from the definitions of λ_i and Λ_i. Using the above inequality, we have, for any $t \geq 2$,

$$\min_{1\leq i\leq t}\mathcal{G}_k(u_i) \leq \frac{\sum_{i=2}^{t}\frac{\lambda_{i+1}}{\Lambda_{i+1}}\mathcal{G}_k(u_i)}{\sum_{i=2}^{t}\frac{\lambda_{i+1}}{\Lambda_{i+1}}} \leq \frac{\sum_{i=2}^{t} i\xi_i + \xi_2 + \frac{t-1}{\eta_k}D^2}{\sum_{i=2}^{t} i} \overset{\text{(d)}}{\leq} \frac{6t-4}{t(t+1)-2}\frac{D^2}{\eta_k} \leq \frac{6D^2}{\eta_k(t+1)},$$

where (d) follows from (13). This implies the number of CG iterations required to ensure $\mathcal{G}_k(u_t) \leq \delta_k$ is at most T_k. □

A.2 Proof of Proposition 3.1

Proof of Proposition 3.1. Since $\mathcal{G}_k(z_k) \le \delta_k$, for any $z \in P$ we have

$$\langle \eta_k \nabla f(x_{k-1}) + (z_k - y_{k-1}), z - z_k \rangle \ge -\eta_k \delta_k. \tag{14}$$

Similarly, taking $z \leftarrow z_k$ yields

$$\langle \eta_{k-1} \nabla f(x_{k-2}) + (z_{k-1} - y_{k-2}), z_k - z_{k-1} \rangle \ge -\eta_{k-1} \delta_{k-1}. \tag{15}$$

By the update y_k, *i.e.*, $y_k = (1-\beta_k) y_{k-1} + \beta_k z_k$, we have $z_{k-1} - y_{k-2} = \frac{1}{1-\beta_{k-1}}(z_{k-1} - y_{k-1})$, and hence (15) can be rewritten as

$$\left\langle \eta_k \nabla f(x_{k-2}) + \frac{\eta_k}{\eta_{k-1}(1-\beta_{k-1})}(z_{k-1} - y_{k-1}), z_k - z_{k-1} \right\rangle \ge -\eta_k \delta_{k-1}. \tag{16}$$

Adding (14) and (16), we obtain

$$\begin{aligned}
&\langle \eta_k \nabla f(x_{k-1}), z - z_{k-1} \rangle + \eta_k \langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k \rangle \\
&\quad + \langle z_k - y_{k-1}, z - z_k \rangle + \frac{\eta_k}{\eta_{k-1}(1-\beta_{k-1})} \langle z_{k-1} - y_{k-1}, z_k - z_{k-1} \rangle \ge -\eta_k(\delta_k + \delta_{k-1}).
\end{aligned}$$

Moreover, applying the identity $2\langle x - y, z - x \rangle = \|y - z\|^2 - \|x - y\|^2 - \|x - z\|^2$ to both terms in the second line on the left-hand side, we obtain

$$\begin{aligned}
&\langle \eta_k \nabla f(x_{k-1}), z - z_{k-1} \rangle + \eta_k \langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k \rangle \\
&\quad + \frac{1}{2}\|y_{k-1} - z\|^2 - \frac{1}{2}\|z_k - y_{k-1}\|^2 - \frac{1}{2}\|z_k - z\|^2 \\
&\quad + \frac{\eta_k}{2\eta_{k-1}(1-\beta_{k-1})}\left(\|z_k - y_{k-1}\|^2 - \|z_{k-1} - y_{k-1}\|^2 - \|z_k - z_{k-1}\|^2\right) \ge -\eta_k(\delta_k + \delta_{k-1}).
\end{aligned} \tag{17}$$

In addition, we have

$$\begin{aligned}
\|z_k - z\|^2 &= \left\| \frac{1}{\beta_k}(y_k - z) - \frac{1-\beta_k}{\beta_k}(y_{k-1} - z) \right\|^2 \\
&\overset{\text{(a)}}{=} \frac{1}{\beta_k}\|y_k - z\|^2 - \frac{1-\beta_k}{\beta_k}\|y_{k-1} - z\|^2 + \frac{1}{\beta_k}\frac{1-\beta_k}{\beta_k}\|y_k - y_{k-1}\|^2 \\
&\overset{\text{(b)}}{=} \frac{1}{\beta_k}\|y_k - z\|^2 - \frac{1-\beta_k}{\beta_k}\|y_{k-1} - z\|^2 + (1-\beta_k)\|z_k - y_{k-1}\|^2,
\end{aligned} \tag{18}$$

where (a) follows from $\|\alpha a + (1-\alpha)b\|^2 = \alpha\|a\|^2 + (1-\alpha)\|b\|^2 - \alpha(1-\alpha)\|a - b\|^2$ for all $\alpha \in \mathbb{R}$ and (b) follows from the update $y_k = (1-\beta_k)y_{k-1} + \beta_k z_k$. Substituting (18) into (17) and rearranging, we obtain

$$\begin{aligned}
&\langle \eta_k \nabla f(x_{k-1}), z_{k-1} - z \rangle - \eta_k(\delta_k + \delta_{k-1}) + \frac{1}{2\beta_k}\|y_k - z\|^2 \\
&\quad + \frac{\eta_k}{2\eta_{k-1}(1-\beta_{k-1})}\left(\|z_{k-1} - y_{k-1}\|^2 + \|z_k - z_{k-1}\|^2\right) \\
&\quad \le \frac{1}{2\beta_k}\|y_{k-1} - z\|^2 + \eta_k \langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k \rangle \\
&\quad - \left(\frac{1}{2} + \frac{1-\beta_k}{2} - \frac{\eta_k}{2\eta_{k-1}(1-\beta_{k-1})} \right) \|z_k - y_{k-1}\|^2.
\end{aligned} \tag{19}$$

For $k \ge 3$, since $\|z_{k-1} - y_{k-1}\|^2 + \|z_k - z_{k-1}\|^2 \ge \frac{1}{2}\|z_k - y_{k-1}\|^2$, we can further simplify (19) to obtain

$$\begin{aligned}
&\langle \eta_k \nabla f(x_{k-1}), z_{k-1} - z \rangle - \eta_k(\delta_k + \delta_{k-1}) + \frac{1}{2\beta_k}\|y_k - z\|^2 \\
&\quad \le \frac{1}{2\beta_k}\|y_{k-1} - z\|^2 + \eta_k \langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k \rangle \\
&\quad - \left(\frac{1}{2} + \frac{1-\beta_k}{2} - \frac{\eta_k}{4\eta_{k-1}(1-\beta_{k-1})} \right) \|z_k - y_{k-1}\|^2 \\
&\quad \overset{\text{(c)}}{\le} \frac{1}{2\beta_k}\|y_{k-1} - z\|^2 + \eta_k \langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k \rangle - \frac{1}{2}\|z_k - y_{k-1}\|^2,
\end{aligned} \tag{20}$$

where (c) follows from $\eta_k \le 2(1-\beta)^2\eta_{k-1}$ in (7). Moreover,

$$\begin{aligned}
&\tau_{k-1}f(x_{k-1}) + \langle \nabla f(x_{k-1}), x_{k-1} - z\rangle \\
&\overset{\text{(d)}}{=} \tau_{k-1}\big(f(x_{k-1}) + \langle \nabla f(x_{k-1}), x_{k-2} - x_{k-1}\rangle\big) + \langle \nabla f(x_{k-1}), z_{k-1} - z\rangle \\
&\overset{(4)}{=} \tau_{k-1}\left(f(x_{k-2}) - \frac{\|\nabla f(x_{k-1}) - \nabla f(x_{k-2})\|^2}{2L_{k-1}}\right) + \langle \nabla f(x_{k-1}), z_{k-1} - z\rangle, \qquad (21)
\end{aligned}$$

where (d) follows from $x_{k-1} - z = z_{k-1} - z + \tau_{k-1}(x_{k-2} - x_{k-1})$ from the update x_k. Combining (20) and (21), we obtain

$$\begin{aligned}
&\eta_k\big(\tau_{k-1}f(x_{k-1}) + \langle \nabla f(x_{k-1}), x_{k-1} - z\rangle - \tau_{k-1}f(x_{k-2}) - (\delta_k + \delta_{k-1})\big) \\
&\le \frac{1}{2\beta_k}\|y_{k-1} - z\|^2 - \frac{1}{2\beta_k}\|y_k - z\|^2 + \eta_k\langle \nabla f(x_{k-1}) - \nabla f(x_{k-2}), z_{k-1} - z_k\rangle \\
&\quad - \frac{1}{2}\|z_k - y_{k-1}\|^2 - \frac{\eta_k\tau_{k-1}}{2L_{k-1}}\|\nabla f(x_{k-1}) - \nabla f(x_{k-2})\|^2. \qquad (22)
\end{aligned}$$

Considering (19) for the case $k = 2$ and using $\beta_1 = 0$, $\tau_1 = 0$, and $x_1 = z_1$, we obtain

$$\begin{aligned}
&\langle \eta_2\nabla f(x_1), z_1 - z\rangle - \eta_2(\delta_2 + \delta_1) + \frac{\eta_2}{2\eta_1}\big(\|z_1 - y_1\|^2 + \|z_2 - z_1\|^2\big) \\
&\le \frac{1}{2\beta_2}\|y_1 - z\|^2 - \frac{1}{2\beta_2}\|y_2 - z\|^2 + \eta_2\langle \nabla f(x_1) - \nabla f(x_0), z_1 - z_2\rangle \\
&\quad - \left(\frac{1}{2} + \frac{1-\beta_2}{2} - \frac{\eta_2}{2\eta_1}\right)\|z_2 - y_1\|^2 \\
&\overset{\text{(e)}}{\le} \frac{1}{2\beta_2}\|y_1 - z\|^2 - \frac{1}{2\beta_2}\|y_2 - z\|^2 + \eta_2\langle \nabla f(x_1) - \nabla f(x_0), z_1 - z_2\rangle - \frac{1}{2}\|z_2 - y_1\|^2, \qquad (23)
\end{aligned}$$

where (e) holds since $\eta_2 \le (1-\beta)\eta_1$ by (6). Replacing the index by i and summing (23) with (22) for $i = 3, \dots, k+1$, and using $\tau_1 = 0$ and $\beta_i = \beta$ for $i \ge 2$, we obtain

$$\begin{aligned}
&\sum_{i=1}^{k}\eta_{i+1}\big(\tau_i f(x_i) + \langle \nabla f(x_i), x_i - z\rangle - \tau_i f(x_{i-1}) - (\delta_{i+1} + \delta_i)\big) \\
&\quad + \frac{\eta_2}{2\eta_1}\big(\|z_1 - y_1\|^2 + \|z_2 - z_1\|^2\big) \\
&\quad \le \frac{1}{2\beta}\|y_1 - z\|^2 - \frac{1}{2\beta}\|y_{k+1} - z\|^2 + \sum_{i=2}^{k+1}\mathcal{D}_i,
\end{aligned}$$

where, for any $i \ge 2$, $\mathcal{D}_i := \eta_i\langle \nabla f(x_{i-1}) - \nabla f(x_{i-2}), z_{i-1} - z_i\rangle - \frac{\eta_i\tau_{i-1}}{2L_{i-1}}\|\nabla f(x_{i-1}) - \nabla f(x_{i-2})\|^2 - \frac{1}{2}\|z_i - y_{i-1}\|^2$. We have the desired result by rearranging it. □

A.3 Proof of Theorem 3.2

The proof follows the same argument as that of [Li and Lan, 2026, Theorem 1]. We include it here for completeness and to make the presentation self-contained.

Proof of Theorem 3.2. We bound $\mathcal{D}_i$ for $i \ge 3$ and obtain

$$\begin{aligned}
\mathcal{D}_i &\overset{\text{(a)}}{\le} \eta_i\|\nabla f(x_{i-1}) - \nabla f(x_{i-2})\|\|z_{i-1} - z_i\| \\
&\quad - \frac{\eta_i\tau_{i-1}}{2L_{i-1}}\|\nabla f(x_{i-1}) - \nabla f(x_{i-2})\|^2 - \frac{1}{2}\|z_i - y_{i-1}\|^2 \\
&\overset{\text{(b)}}{\le} \frac{\eta_i L_{i-1}}{2\tau_{i-1}}\|z_{i-1} - z_i\|^2 - \frac{1}{2}\|z_i - y_{i-1}\|^2 \\
&\overset{\text{(c)}}{\le} \frac{\eta_i L_{i-1}(1-\beta)^2}{\tau_{i-1}}\|z_{i-1} - y_{i-2}\|^2 - \left(\frac{1}{2} - \frac{\eta_i L_{i-1}}{\tau_{i-1}}\right)\|z_i - y_{i-1}\|^2 \\
&\overset{\text{(d)}}{\le} \frac{1}{4}\|z_{i-1} - y_{i-2}\|^2 - \frac{1}{4}\|z_i - y_{i-1}\|^2,
\end{aligned}$$

where (a) follows from the Cauchy–Schwarz inequality, (b) follows from Young's inequality, (c) follows from $\|z_{i-1} - z_i\|^2 \leq 2(1-\beta)^2\|z_{i-1} - y_{i-2}\|^2 + 2\|z_i - y_{i-1}\|^2$, and (d) follows from $\eta_i \leq \frac{\tau_{i-1}}{4L_{i-1}}$ due to (7). Moreover, it holds that

$$\begin{aligned}
\mathcal{D}_2 &\overset{\text{(e)}}{\leq} \eta_2 L_1 \|x_1 - x_0\| \|z_1 - z_2\| - \frac{1}{2}\|z_2 - y_1\|^2 \\
&\overset{\text{(f)}}{\leq} \eta_2 L_1 \|z_1 - y_0\| (\|z_1 - y_0\| + \|z_2 - y_0\|) - \frac{1}{2}\|z_2 - y_0\|^2 \\
&\overset{\text{(g)}}{\leq} \frac{5\eta_2 L_1}{4}\|z_1 - z_0\|^2 + \left(\eta_2 L_1 - \frac{1}{2}\right)\|z_2 - y_0\|^2 \\
&\overset{\text{(h)}}{\leq} \frac{5\eta_2 L_1}{4}\|z_1 - z_0\|^2 - \frac{1}{4}\|z_2 - y_1\|^2,
\end{aligned}$$

where (e) follows from the Cauchy–Schwarz inequality and (3), (f) follows from the triangle inequality, $x_1 = z_1$, and $x_0 = y_0 = z_0$ due to $\tau_1 = \beta_1 = 0$, (g) follows from Young's inequality, and (h) follows from $\eta_2 \leq \frac{1}{4L_1}$ due to (6). Substituting these inequalities into (8) yields

$$\begin{aligned}
&\sum_{i=1}^{k} \eta_{i+1}\left(\tau_i f(x_i) + \langle \nabla f(x_i), x_i - z\rangle - \tau_i f(x_{i-1}) - (\delta_{i+1} + \delta_i)\right) \\
&\quad \leq \frac{1}{2\beta}\|y_1 - z\|^2 - \frac{1}{2\beta}\|y_{k+1} - z\|^2 + \left(\frac{5\eta_2 L_1}{4} - \frac{\eta_2}{2\eta_1}\right)\|z_1 - z_0\|^2 - \frac{1}{4}\|z_{k+1} - y_k\|^2.
\end{aligned}$$

Taking $z = x^*$, using convexity of f (so that $\langle \nabla f(x_i), x_i - x^*\rangle \geq f(x_i) - f(x^*)$), and using the update x_k, we obtain

$$\begin{aligned}
&\sum_{i=1}^{k-1}\left((\tau_i + 1)\eta_{i+1} - \tau_{i+1}\eta_{i+2}\right)(f(x_i) - f(x^*)) + (\tau_k + 1)\eta_{k+1}(f(x_k) - f(x^*)) \\
&\quad \leq \tau_1\eta_2(f(x_0) - f(x^*)) + \frac{1}{2\beta}\|y_1 - x^*\|^2 + \left(\frac{5\eta_2 L_1}{4} - \frac{\eta_2}{2\eta_1}\right)\|z_1 - z_0\|^2 \\
&\quad\quad - \frac{1}{2\beta}\|y_{k+1} - x^*\|^2 - \frac{1}{4}\|z_{k+1} - y_k\|^2 + \sum_{i=1}^{k}\eta_{i+1}(\delta_{i+1} + \delta_i) \\
&\quad \overset{\text{(i)}}{\leq} \frac{1}{2\beta}\|z_0 - x^*\|^2 + \left(\frac{5\eta_2 L_1}{4} - \frac{\eta_2}{2\eta_1}\right)\|z_1 - z_0\|^2 - \frac{1}{2\beta}\|y_{k+1} - x^*\|^2 - \frac{1}{4}\|z_{k+1} - y_k\|^2 \\
&\quad\quad + \sum_{i=1}^{k}\eta_{i+1}(\delta_{i+1} + \delta_i),
\end{aligned} \tag{24}$$

where (i) follows from $\tau_1 = 0$ and $y_1 = z_0$. By (7), we have $\eta_i \leq \frac{\tau_{i-2}+1}{\tau_{i-1}}\eta_{i-1}$, equivalently, $(\tau_{i-2} + 1)\eta_{i-1} - \eta_i\tau_{i-1} \geq 0$. Therefore, letting $\mathcal{S}_k := \sum_{i=1}^{k}\eta_{i+1}(\delta_{i+1} + \delta_i)$, we have

$$f(x_k) - f(x^*) \leq \frac{1}{(\tau_k + 1)\eta_{k+1}}\left(\frac{1}{2\beta}\|z_0 - x^*\|^2 + \left(\frac{5\eta_2 L_1}{4} - \frac{\eta_2}{2\eta_1}\right)\|z_1 - z_0\|^2 + \mathcal{S}_k\right).$$

Furthermore, define

$$\overline{x}_k = \frac{\sum_{i=1}^{k-1}\left((\tau_i + 1)\eta_{i+1} - \tau_{i+1}\eta_{i+2}\right)x_i + (\tau_k + 1)\eta_{k+1}x_k}{\sum_{i=2}^{k+1}\eta_i}.$$

Since $\sum_{i=1}^{k-1}\left((\tau_i + 1)\eta_{i+1} - \tau_{i+1}\eta_{i+2}\right) + (\tau_k + 1)\eta_{k+1} = \sum_{i=2}^{k+1}\eta_i$, we obtain

$$f(\overline{x}_k) - f(x^*) \leq \frac{1}{\sum_{i=2}^{k+1}\eta_i}\left(\frac{1}{2\beta}\|z_0 - x^*\|^2 + \left(\frac{5\eta_2 L_1}{4} - \frac{\eta_2}{2\eta_1}\right)\|z_1 - z_0\|^2 + \mathcal{S}_k\right).$$

All iterates x_k, y_k, and z_k belong to the compact set P, so they are bounded. □

A.4 Proof of Corollary 3.3

Proof of Corollary 3.3. The stepsize η_k satisfies (6) and (7) from the definition of τ_k and β. By induction, we show that $\eta_k \geq \frac{k}{12\hat{L}_{k-1}}$ for all $k \geq 2$ (its proof is given below). By substituting this bound into Theorem 3.2, we have

$$f(x_k) - f(x^*) \leq \frac{12\hat{L}_k}{k(k+1)}(2\,\mathcal{E} + 2\,\mathcal{S}_k) \tag{25}$$

and

$$f(\overline{x}_k) - f(x^*) \leq \frac{1}{\sum_{i=2}^{k+1} \frac{i}{6\hat{L}_{i-1}}}(2\,\mathcal{E} + 2\,\mathcal{S}_k). \tag{26}$$

We show $\eta_k \leq \frac{k}{2}\eta_2$ for $k \geq 2$ by induction. From the definition of η_k, the base cases $k = 2$ and $k = 3$ hold trivially. Suppose that the claim holds for some $k \geq 3$. Then, from the definition of η_{k+1}, we have

$$\eta_{k+1} \leq \frac{k+1}{k}\eta_k \leq \frac{k+1}{k} \cdot \frac{k}{2}\eta_2 = \frac{k+1}{2}\eta_2.$$

Thus, we have $\eta_k \leq \frac{k}{2}\eta_2$ for all $k \geq 2$. Therefore, we can bound $\mathcal{S}_k = \sum_{i=1}^{k} \eta_{i+1}(\delta_{i+1} + \delta_i)$ as

$$\begin{aligned}\mathcal{S}_k &\overset{\text{(a)}}{\leq} \sum_{i=1}^{k} \frac{i+1}{2}\eta_2 \left(\frac{D_0^2}{(i+1)^{1+\theta}(i+2)} + \frac{D_0^2}{i^{1+\theta}(i+1)} \right) \leq \frac{\eta_2 D_0^2}{2} \sum_{i=1}^{k} \left(\frac{1}{(i+1)^{1+\theta}} + \frac{1}{i^{1+\theta}} \right) \\ &\overset{\text{(b)}}{\leq} \frac{\eta_2 D_0^2}{2} \left(1 + \int_1^{k+1} \frac{2}{x^{1+\theta}} dx \right) = \frac{\eta_2 D_0^2}{2} \left(1 + \frac{2}{\theta}(1 - (k+1)^{-\theta}) \right) \leq \frac{\eta_2 D_0^2 (2+\theta)}{2\theta},\end{aligned}$$

where (a) follows from $\eta_{i+1} \leq \frac{i+1}{2}\eta_2$ and (b) follows from the integral test. By substituting this bound into (25) and (26) and using $\eta_2 \leq (1-\beta)\eta_1$, we obtain the desired result. □

The proof of $\eta_k \geq \frac{k}{12\hat{L}_{k-1}}$ follows the same argument as that of [Li and Lan, 2026, Corollary 1]. We include it here for completeness and to make the presentation self-contained.

Proof of $\eta_k \geq \frac{k}{12\hat{L}_{k-1}}$. We prove $\eta_k \geq \frac{k}{12\hat{L}_{k-1}}$ by induction on k. Recall the definition of $\hat{L}_k$ as follows:

$$\hat{L}_k := \max\left\{ \frac{1}{4(1-\beta)\eta_1},\ \max_{1 \leq i \leq k} L_i \right\}.$$

For $k = 2$, by the definition of η_2 we have

$$\eta_2 = \min\left\{ (1-\beta)\eta_1,\ \frac{1}{4L_1} \right\} = \frac{1}{4\hat{L}_1}.$$

For $k = 3$,

$$\eta_3 = \min\left\{ \eta_2,\ \frac{1}{4L_2} \right\} = \min\left\{ \frac{1}{4\hat{L}_1},\ \frac{1}{4L_2} \right\} = \frac{1}{4\hat{L}_2} \geq \frac{3}{12\hat{L}_2}.$$

Now assume that for some $k \geq 3$, $\eta_k \geq \frac{k}{12\hat{L}_{k-1}}$. Then, using the stepsize rule,

$$\eta_{k+1} = \min\left\{ \frac{k+1}{k}\eta_k,\ \frac{k}{8L_k} \right\} \geq \min\left\{ \frac{k+1}{12\hat{L}_{k-1}},\ \frac{k}{8L_k} \right\} \geq \min\left\{ \frac{k+1}{12},\ \frac{k}{8} \right\} \cdot \frac{1}{\hat{L}_k}.$$

Since $k \geq 3$ implies $\frac{k}{8} \geq \frac{k+1}{12}$, the minimum equals $\frac{k+1}{12}$, and hence

$$\eta_{k+1} \geq \frac{k+1}{12\hat{L}_k},$$

which completes the induction. □

A.5 Proof of Corollary 3.5

Proof of Corollary 3.5. Using $\eta_k \leq \frac{k}{2}\eta_2$ and $\delta_k = \frac{D_0^2}{Nk}$, we now bound $\mathcal{S}_k = \sum_{i=1}^k \eta_{i+1}(\delta_{i+1} + \delta_i)$ as follows:

$$\mathcal{S}_N \leq \frac{\eta_2 D_0^2}{2N} \sum_{i=1}^N \left(1 + \frac{i+1}{i}\right) \overset{\text{(a)}}{\leq} \eta_2 D_0^2 \left(1 + \frac{1+\log N}{2N}\right) \leq \frac{3}{2}\eta_2 D_0^2,$$

where (a) follows from the integral test. Therefore, by substituting this bound into (25) with $k = N$ and using $\eta_2 \leq (1-\beta)\eta_1$, we obtain the desired result. □

A.6 Proof of Corollary 3.6

Proof of Corollary 3.6. Using $2(1-\beta)^2 \geq 4/3$ due to $\beta \in (0, 1 - \sqrt{6}/3]$ and the setting of η_k, we find that the stepsize η_k satisfies (7). By the definition of τ_k, we have for any $k \geq 3$,

$$\tau_k = \tau_2 + \sum_{i=3}^k \left(\frac{\alpha}{2} + \frac{2(1-\alpha)\eta_i L_{i-1}}{\tau_{i-1}}\right) \overset{\text{(a)}}{\leq} 1 + \sum_{i=3}^k \frac{1}{2} = \frac{k}{2}, \tag{27}$$

where (a) follows from $\tau_2 = 1$ and $\eta_i \leq \frac{\tau_{i-1}}{4L_{i-1}}$. Therefore, for any $k \geq 3$, we also obtain

$$\frac{\tau_{k-1}+1}{\tau_k} \overset{\text{(b)}}{\geq} \frac{\tau_k + 1/2}{\tau_k} \overset{\text{(c)}}{\geq} \frac{k+1}{k}, \tag{28}$$

where (b) follows from $\tau_k \leq \tau_{k-1} + 1/2$ due to the definitions of η_k and τ_k and (c) follows from (27). These inequalities imply that $\tau_k \geq 1 + \frac{\alpha(k-2)}{2}$. By induction, we can show $\eta_k \geq \frac{3+\alpha(k-3)}{12\hat{L}_{k-1}}$ for all $k \geq 2$ (its proof is given below). We bound $\mathcal{S}_k$ as follows:

$$\begin{aligned}
\mathcal{S}_k &\overset{\text{(d)}}{\leq} \sum_{i=1}^k \frac{i+1}{8L_i}(\delta_{i+1} + \delta_i) = \sum_{i=1}^k \frac{i+1}{8L_i}\left(\frac{D_0^2}{(i+1)^{1+\theta}(i+2)} + \frac{D_0^2}{i^{1+\theta}(i+1)}\right) \\
&\leq \frac{D_0^2}{8\underline{L}_k} \sum_{i=1}^k \left(\frac{1}{(i+1)^{1+\theta}} + \frac{1}{i^{1+\theta}}\right) \leq \frac{D_0^2}{8\underline{L}_k}\left(1 + \frac{2}{\theta}\right),
\end{aligned}$$

where (d) follows from $\eta_2 \leq \frac{1}{4L_1}$, $\eta_{i+1} \leq \frac{\tau_i}{4L_i}$ for $i \geq 2$, and (27). Substituting this into Theorem 3.2, we obtain the desired results. □

The proof of $\eta_k \geq \frac{3+\alpha(k-3)}{12\hat{L}_{k-1}}$ follows the same argument as that of [Li and Lan, 2026, Corollary 2]. We include it here for completeness and to make the presentation self-contained.

Proof of $\eta_k \geq \frac{3+\alpha(k-3)}{12\hat{L}_{k-1}}$. We prove the claim by induction on k. For $k = 2$, by the definition of η_2, we have

$$\eta_2 = \min\left\{(1-\beta)\eta_1, \frac{1}{4L_1}\right\} = \frac{1}{4\hat{L}_1}.$$

For $k = 3$, we have

$$\eta_3 = \min\left\{\eta_2, \frac{1}{4L_2}\right\} = \min\left\{\frac{1}{4\hat{L}_1}, \frac{1}{4L_2}\right\} = \frac{1}{4\hat{L}_2} \geq \frac{3}{12\hat{L}_2}.$$

Now assume that for some $k \geq 3$, $\eta_k \geq \frac{3+\alpha(k-3)}{12\hat{L}_{k-1}}$. Then, using the stepsize rule, we have

$$\begin{aligned}
\eta_{k+1} &= \min\left\{\frac{4}{3}\eta_k, \frac{\tau_{k-1}+1}{\tau_k}\eta_k, \frac{\tau_k}{4L_k}\right\} \\
&\overset{\text{(a)}}{\geq} \min\left\{\frac{k+1}{k}\eta_k, \frac{1+\alpha(k-2)/2}{4L_k}\right\} \\
&\geq \min\left\{\frac{k+1}{k}\cdot\frac{3+\alpha(k-3)}{12\hat{L}_{k-1}}, \frac{2+\alpha(k-2)}{8L_k}\right\} \geq \frac{3+\alpha(k-2)}{12\hat{L}_k},
\end{aligned}$$

where (a) follows from (28). Since $\alpha \in [0,1]$ and $k \geq 3$, both terms in the minimum are at least $\frac{3+\alpha(k-2)}{12\hat{L}_k}$, and hence

$$\eta_{k+1} \geq \frac{3+\alpha(k-2)}{12\hat{L}_k},$$

which completes the induction. □

A.7 Proof of Theorem 4.1

Proof of Theorem 4.1. We prove the claim by induction. The base case $s = 0$ holds trivially. Suppose that the claim holds for some $s \geq 0$, *i.e.*, $f(w_s) - f(x^*) \leq \frac{\phi_0}{2^s}$. The sequence$\{w_s\}$ is bounded because $\{x_k\}$ is bounded from Theorem 3.2, $w_s = x_N$ for some N, and P is compact. Using the local strong convexity of f, we have

$$\|w_s - x^*\|^2 \leq \frac{2}{\mu}(f(w_s) - f(x^*)) \leq \frac{2\phi_0}{2^s\mu},$$

where μ is the local strong convexity parameter of f. Let $D_0^2 := \frac{2\phi_0}{2^s\mu}$, an upper bound on $\|w_s - x^*\|^2$. The next stage uses $\delta_k = (1-\beta)D_0^2/(4N\eta_k)$. Since the stepsize rule gives $\eta_{i+1} \leq 2\eta_i$, we have

$$\mathcal{S}_N = \frac{(1-\beta)D_0^2}{4N}\sum_{i=1}^{N}\left(1 + \frac{\eta_{i+1}}{\eta_i}\right) \leq \frac{3(1-\beta)D_0^2}{4}.$$

With $x_0 = y_0 = z_0 = w_s$, $x_N = w_{s+1}$, and the first-iteration line search, we have $\mathcal{E} \leq D_0^2/(2\beta)$ and $\hat{L}_N \leq 5\gamma L/[8(1-\beta)]$. Substituting these bounds into (25), we obtain

$$\begin{aligned} f(w_{s+1}) - f(x^*) &\leq \frac{15\gamma L D_0^2}{4N(N+1)}\left(\frac{2}{\beta(1-\beta)} + 3\right) \\ &= \frac{15\gamma L}{4N(N+1)}\left(\frac{2}{\beta(1-\beta)} + 3\right)\cdot\frac{2\phi_0}{2^s\mu} \leq \frac{\phi_0}{2^{s+1}}, \end{aligned}$$

Thus, by induction, we have $f(w_s) - f(x^*) \leq \frac{\phi_0}{2^s}$ for all $s \geq 0$.

Next, let $S := \lceil \log_2 \max\{\phi_0/\epsilon, 1\}\rceil$. Excluding rejected initialization trials, the number of gradient computations required to achieve $f(w_s) - f(x^*) \leq \epsilon$ is $O(NS)$, which is bounded by $O(N_0 \log_2\lceil \max\{\phi_0/\epsilon, 1\}\rceil)$. Finally, we analyze the total number of linear minimizations. Let $T_{i,k}$ be the number of CG iterations performed to solve (2) up to the accuracy δ_k at the k-th outer iteration of the i-th stage. From T_k in Theorem 2.1, it holds that

$$T_{i,k} \leq \frac{6D^2}{\eta_k\delta_k} + 1 \overset{\text{(a)}}{\leq} \frac{6\mu D^2 2^i N}{(1-\beta)\phi_0} + 1,$$

where (a) follows from the definition of δ_k in Algorithm 3. Therefore, the total number of CG iterations performed in the first S stages is at most

$$\begin{aligned} \sum_{s=1}^{S}\sum_{k=1}^{N} T_{s,k} &\leq \sum_{s=1}^{S}\sum_{k=1}^{N}\frac{6\mu D^2 2^s N}{(1-\beta)\phi_0} + NS \\ &= \frac{6\mu D^2 N^2}{(1-\beta)\phi_0}\sum_{s=1}^{S} 2^s + NS \\ &\leq \frac{6\mu D^2 N^2}{(1-\beta)\phi_0} 2^{S+1} + NS \\ &\overset{\text{(b)}}{\leq} \frac{24\mu D^2 N^2}{(1-\beta)\epsilon} + NS \end{aligned}$$

where (b) follows from the definition of S. This inequality is bounded by $O(LD^2/\epsilon + N_0 \log_2\lceil \max\{\phi_0/\epsilon, 1\}\rceil)$. □

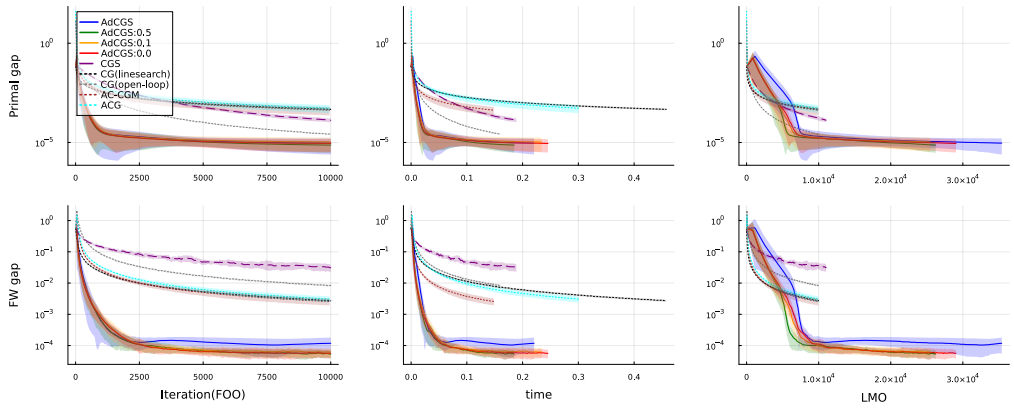


Figure 5: Least squares regression over the simplex with $(m, n) = (1000, 200)$.

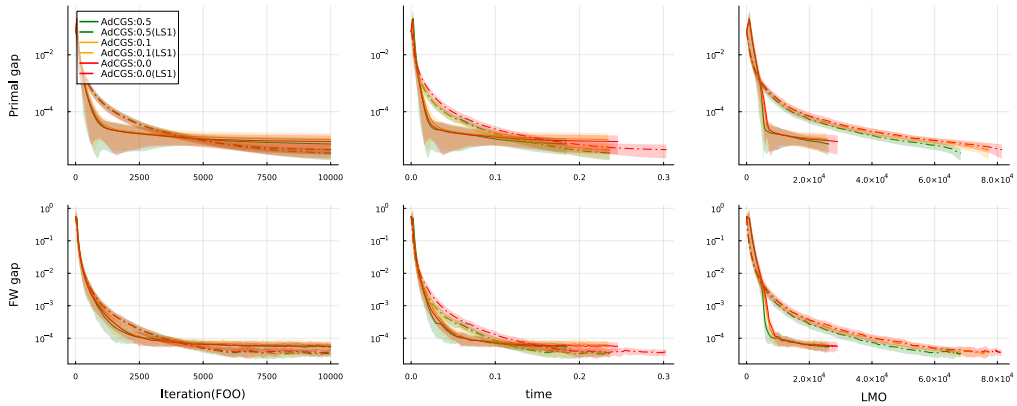


Figure 6: Least squares regression over the simplex with $(m, n) = (1000, 200)$.

B Additional Experimental Results

B.1 Least Squares Regression over the Simplex

We provide additional experimental results for least squares regression over the simplex, including results for the first-iteration line-search strategy in Algorithm 2. In Figure 5, the proposed method achieves the best performance. In particular, AdCGS:0.5 outperforms both the baseline methods and the other AdCGS variants. In Figure 6, the first-iteration line-search variants remain stable, whereas AdCGS with $\alpha = 1.0$ is the exception.

B.2 ℓ_p Loss Regression over the ℓ_2-Ball

We provide additional experimental results for ℓ_p loss regression over the ℓ_2-ball. In Figure 7, ACG attains good objective values but exhibits instability, whereas the proposed methods and AC-FGM perform consistently well.

B.3 Logistic Regression

We provide additional experimental results for logistic regression. Overall, the results are favorable: AdCGS:0.1 performs best on Figures 8-10.

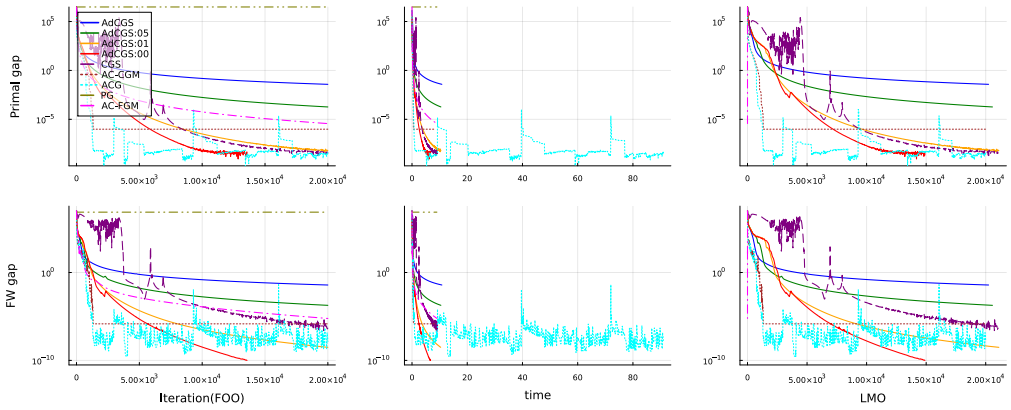


Figure 7: ℓ_p loss regression over the cpusmall dataset with $(m, n) = (8192, 12)$.

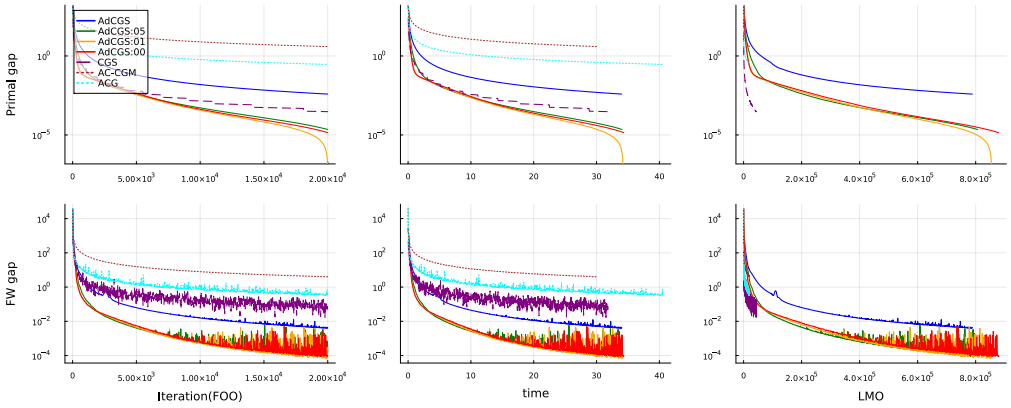


Figure 8: Logistic regression over the a9a dataset with $(m, n) = (32561, 123)$.

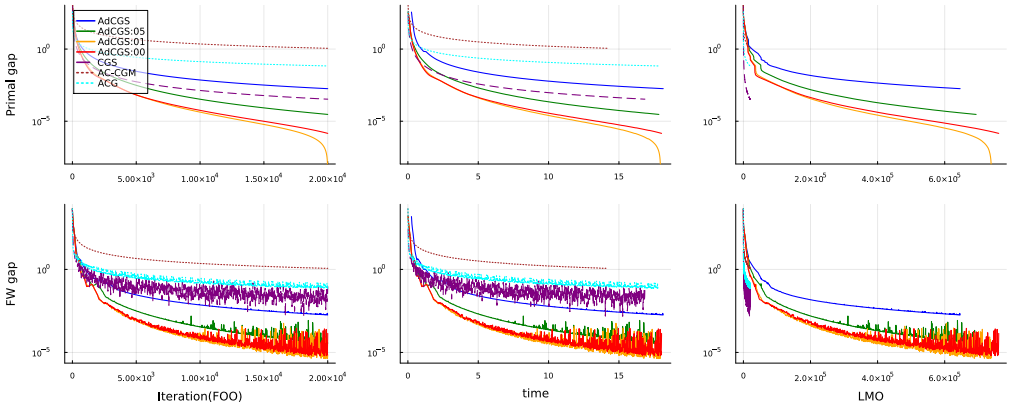


Figure 9: Logistic regression over the phishing dataset with $(m, n) = (11055, 68)$.

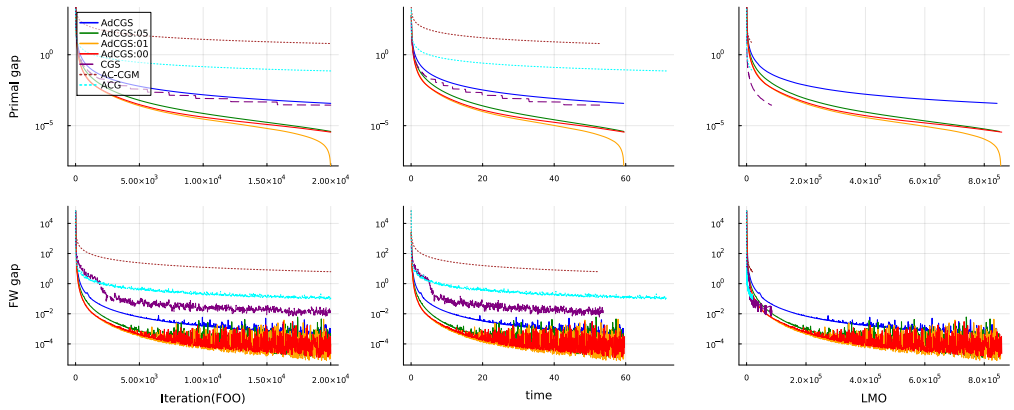


Figure 10: Logistic regression over the w8a dataset with $(m, n) = (49749, 300)$.

B.4 Poisson Regression over Positive Semidefinite Matrices with a Trace Constraint

We consider the following Poisson negative log-likelihood problem over positive semidefinite matrices with a trace constraint:

$$\min_{X \succeq 0, \operatorname{Tr}(X)=\tau} \quad \frac{1}{m} \sum_{i=1}^{m} (\mu_i(X) - y_i \log \mu_i(X)),$$

where $\mathbb{S}^d$ denotes real symmetric matrices, $y_i \geq 0$ is the observed count, $\tau > 0$ is the prescribed trace, and $\mu_i(X) = q_i^\top X q_i + b_i$ for measurement vectors $q_i \in \mathbb{R}^d$ and background intensities $b_i > 0$. Since $\mu_i(X) \geq b_i > 0$, f is convex and smooth on the feasible set, though its global smoothness bound can be conservative. The LMO returns $\tau v v^\top$, where v is a unit eigenvector for the smallest eigenvalue of the gradient matrix; Euclidean projection requires a full eigendecomposition and simplex projection of the eigenvalues. All methods start at $X_0 = (\tau/d)I$.

We set $(d, m) = (250, 2000)$ and $\tau = 1$. Figure 11 shows convergence under a common outer-iteration budget. Curves and shaded bands show the mean and standard deviation over 10 instances. The AdCGS variants achieve final primal gaps comparable to AC-FGM in less time.

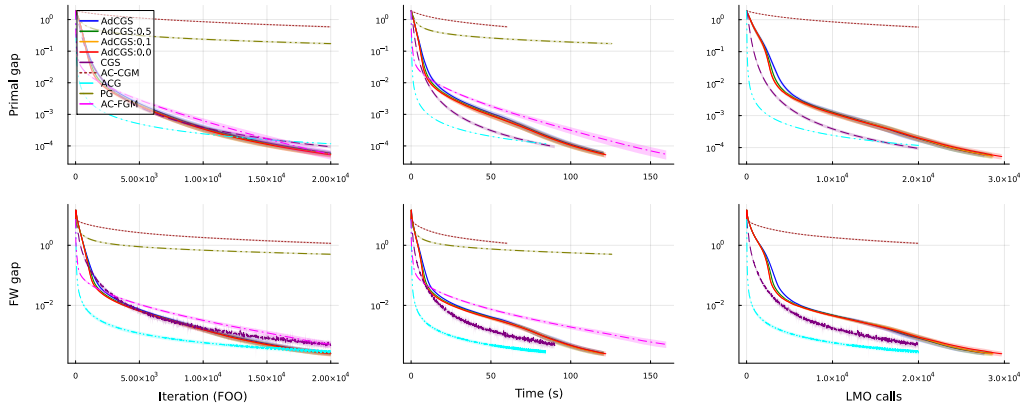


Figure 11: Poisson regression over positive semidefinite matrices with $(d, m) = (250, 2000)$.

B.5 LMO Calls per Gradient Evaluation

Tables 1-4 report total LMO calls divided by total gradient evaluations, measuring the average inner CG iterations per gradient evaluation. We report each setting separately because relative oracle costs depend on the problem.

Table 1: LMO calls per gradient evaluation for least squares regression.

Setting	AdCGS:1.0	AdCGS:0.5	AdCGS:0.1	AdCGS:0.0	CGS
$m = 1000, n = 200$	3.5	2.6	2.5	2.9	1.1
$m = 2500, n = 500$	2.5	2.3	2.7	2.9	1.0

Table 2: LMO calls per gradient evaluation for ℓ_p loss regression.

Dataset	AdCGS:1.0	AdCGS:0.5	AdCGS:0.1	AdCGS:0.0	CGS
cpusmall	1.0	1.0	1.1	1.1	1.1
YearPredictionMSD	1.8	1.8	2.2	2.4	1.2

Table 3: LMO calls per gradient evaluation for logistic regression.

Dataset	AdCGS:1.0	AdCGS:0.5	AdCGS:0.1	AdCGS:0.0	CGS
a9a	39	40	43	44	2.2
gisette	50	50	50	50	10
phishing	32	35	37	38	1.0
w8a	42	43	43	43	4.2

Table 4: LMO calls per gradient evaluation for Poisson regression.

Setting	AdCGS:1.0	AdCGS:0.5	AdCGS:0.1	AdCGS:0.0	CGS
$d = 250, m = 2000$	1.4	1.4	1.4	1.5	1.0

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our claims are written in the abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses several limitations of the proposed method in Section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Theorems 2.1, 3.2, and 4.1, Proposition 3.1, and Corollaries 3.3, 3.5, and 3.6 declare the assumptions in the first several lines of their statements. The proofs are provided in Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 5 and Appendix B provide detailed information about the experimental setup, including the datasets used, the implementation details, and the evaluation metrics. This information is sufficient to reproduce the main experimental results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code to reproduce the results is included as part of the supplementary material. The datasets are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The hyperparameters and type of optimizer are included in the experiment section (Section 5) and Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the mean and standard deviation over 20 independently generated random instances (random seeds 1–20) for Figures 1, 2, 5, and 6, and over 10 instances (random seeds 1–10) for Figure 11.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide information on the type of CPU used for the experiments, along with the memory requirement in the experiment section (Section 5).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We ensure that the research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: There is no societal impact of the work performed because it is foundational research and not tied to particular applications.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release pre-trained language models, image generators, or scraped datasets that have a high risk of misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have properly cited the code and datasets used in the paper in the experiment section (Section 5).

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not introduce new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Such experiments are not relevant to this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were used only for writing, editing, formatting, and basic code assistance. Their use did not affect the core methodology, scientific rigor, or originality of the research. The authors reviewed and verified all LLM-assisted outputs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.