An Accelerated Gradient Method for Convex Smooth Simple Bilevel Optimization

Jincheng Cao* Ruichen Jiang* Erfan Yazdandoost Hamedani† Aryan Mokhtari*

Abstract

In this paper, we focus on simple bilevel optimization problems, where we minimize a convex smooth objective function over the optimal solution set of another convex smooth constrained optimization problem. We present a novel bilevel optimization method that locally approximates the solution set of the lower-level problem using a cutting plane approach and employs an accelerated gradient-based update to reduce the upper-level objective function over the approximated solution set. We measure the performance of our method in terms of suboptimality and infeasibility errors and provide non-asymptotic convergence guarantees for both error criteria. Specifically, when the feasible set is compact, we show that our method requires at most $\mathcal{O}(\max\{1/\sqrt{\epsilon_f}, 1/\epsilon_g\})$ iterations to find a solution that is ϵ_f-suboptimal and ϵ_g-infeasible. Moreover, under the additional assumption that the lower-level objective satisfies the r-th Hölderian error bound, we show that our method achieves an iteration complexity of $\mathcal{O}(\max\{\epsilon_f^{-\frac{2r-1}{2r}}, \epsilon_g^{-\frac{2r-1}{2r}}\})$, which matches the optimal complexity of single-level convex constrained optimization when $r = 1$.

*Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA {jinchengcao@utexas.edu, rjiang@utexas.edu, mokhtari@austin.utexas.edu}

†Department of Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, USA {erfany@arizona.edu}

1 Introduction

In this paper, we investigate a class of bilevel optimization problems known as simple bilevel optimization, in which we aim to minimize an upper-level objective function over the solution set of a corresponding lower-level problem. Recently this class of problems has gained great attention due to their board applications in continual learning [BMK20], hyper-parameter optimization [FFSGP18; SCHB19], meta-learning [RFKL19; BHTV18], and over-parameterized machine learning [JAMH23; CJAYM24; SBY23]. Specifically, we focus on the following bilevel optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; f(\mathbf{x}) \qquad \text{s.t.} \quad \mathbf{x} \in \operatorname*{argmin}_{\mathbf{z}\in\mathcal{Z}} \; g(\mathbf{z}), \tag{1}$$

where $\mathcal{Z}$ is a convex set and $f, g : \mathbb{R}^n \to \mathbb{R}$ are convex and continuously differentiable functions on an open set containing $\mathcal{Z}$. We assume that the lower-level objective function g is convex but may not be strongly convex. Hence, the lower-level problem may have multiple optimal solutions. Throughout the paper, we will use $\mathbf{x}^*$ to denote an optimal solution of problem (1). Further, we define $f^* \triangleq f(\mathbf{x}^*)$ and $g^* \triangleq g(\mathbf{x}^*)$, which represent the optimal value of problem (1) and the optimal value of the lower-level objective g, respectively. This class of problems is often referred to as the "simple bilevel problem" in the literature [DDD10; DP20; SVZ21] to distinguish it from general settings where the lower-level problem is parameterized by some upper-level variables.

The main challenge in solving problem (1) arises from the fact that the feasible set, i.e., the optimal solution set of the lower-level problem, lacks a simple characterization and is not explicitly provided. Consequently, direct application of projection-based or projection-free methods is infeasible, as projecting onto or solving a linear minimization problem over such an implicitly defined feasible set is intractable. Instead, our approach begins by constructing an approximation set that possesses specific properties, serving as a surrogate for the true feasible set of the bilevel problem. In Section 3, we delve into the details of how such a set is constructed. By using this technique and building upon the idea of the projected accelerated gradient method, we establish the best-known complexity bounds for solving problem (1).

To provide a clearer context for our results, note that the best-known complexity bound for achieving an ϵ-accurate solution in single-level convex constrained optimization problems is $\mathcal{O}(\epsilon^{-0.5})$, as demonstrated in [BT09]. This bound is optimal and was achieved using the accelerated proximal method or FISTA (Fast Iterative Shrinkage-Thresholding Algorithm), which plays a canonical role in the development of our algorithm as well.

While the literature on bilevel optimization is not as extensive as that for single-level optimization, there have been recent non-asymptotic results for solving this class of problems, which we summarize in Table 1. Specifically, these results aim to establish convergence rates on the *infeasibility gap* $g(\mathbf{x}_k) - g^*$ and the *suboptimality gap* $f(\mathbf{x}_k) - f^*$ after k iterations. Kaushik and Yousefian [KY21] demonstrated that an iterative regularization-based method achieves a convergence rate of $\mathcal{O}(1/k^{0.5-b})$ in terms of suboptimality and a rate of $\mathcal{O}(1/k^b)$ in terms of infeasibility, where $b \in (0, 0.5)$ is a user-defined parameter. Therefore, if we set $b = 0.25$ to balance these two rates, it would require an iteration complexity of $O(\max\{1/\epsilon_f^4, 1/\epsilon_g^4\})$ to find a solution that is ϵ_f-optimal and ϵ_g-infeasible. Later, the Bi-Sub-Gradient (Bi-SG) algorithm was proposed by Merchav and Sabach [MS23] to address convex simple bilevel optimization problems with nonsmooth upper-level objective functions. They showed convergence rates of $\mathcal{O}(1/k^{1-\alpha})$ and $\mathcal{O}(1/k^\alpha)$ in terms of suboptimality and infeasibility, respectively, where $\alpha \in (0.5, 1)$ serves as a hyper-parameter. This implies that if we balance the rates by setting $\alpha = 0.5$, the iteration complexity of Bi-SG is $\mathcal{O}(\max\{1/\epsilon_f^2, 1/\epsilon_g^2\})$. Additionally, Shen, Ho-Nguyen, and Kılınç-Karzan [SHK23] introduced a

Table 1: Non-asymptotic results on simple bilevel optimization. (①: with a first-order Hölderian error bound assumption on g; ⓡ: with an rth-order ($r \geq 1$) Hölderian error bound assumption on g; Ⓐ: with an additional assumption implying that the projection onto the sublevel set of f is easy to compute.)

References	Upper level	Lower level		Convergence	
	Objective f	Objective g	Feasible set $\mathcal{Z}$	Upper level	Lower level
a-IRG [KY21]	Convex, Lipschitz	Convex, Lipschitz	Closed	$\mathcal{O}(1/\epsilon_f^4)$	$\mathcal{O}(1/\epsilon_g^4)$
Bi-SG [MS23]	Convex, Nonsmooth	Convex, Composite	Closed	$\mathcal{O}(1/\epsilon_f^{\frac{1}{1-\alpha}})$	$\mathcal{O}(1/\epsilon_g^{\frac{1}{\alpha}}), \alpha \in (0.5, 1)$
SEA [SHK23]	Convex	Convex, Smooth	Compact	$\mathcal{O}(1/\epsilon_f^2)$	$\mathcal{O}(1/\epsilon_g^2)$
CG-BiO [JAMH23]	Convex, Smooth	Convex, Smooth	Compact	$\mathcal{O}(1/\epsilon_f)$	$\mathcal{O}(1/\epsilon_g)$
R-APM [SBY23]	Convex, Smooth	Convex, Composite	Closed	$\mathcal{O}(1/\epsilon_f)$	$\mathcal{O}(1/\epsilon_g)$
AGM-BiO (Ours)	Convex, Smooth	Convex, Smooth	Compact	$\mathcal{O}(1/\epsilon_f^{0.5})$	$\mathcal{O}(1/\epsilon_g)$
R-APM ① [SBY23]	Convex, Smooth	Convex, Composite	Closed	$\mathcal{O}(1/\epsilon_f^{0.5})$	$\mathcal{O}(1/\epsilon_g^{0.5})$
Bisec-BiO Ⓐ [WSJ24]	Convex, Composite	Convex, Composite	Closed	$\tilde{\mathcal{O}}(\max\{1/\epsilon_f^{0.5}, 1/\epsilon_g^{0.5}\})$	
AGM-BiO ① (Ours)	Convex, Smooth	Convex, Smooth	Closed	$\mathcal{O}(1/\epsilon_f^{0.5})$	$\tilde{\mathcal{O}}(1/\epsilon_g^{0.5})$
PB-APG ⓡ [CSJW24]	Convex, Composite	Convex, Composite	Compact	$\mathcal{O}(1/\epsilon_f^{0.5r}) + \mathcal{O}(1/\epsilon_g^{0.5})$	
AGM-BiO ⓡ (Ours)	Convex, Smooth	Convex, Smooth	Closed	$\tilde{\mathcal{O}}(1/\epsilon_f^{\frac{2r-1}{2r}})$	$\tilde{\mathcal{O}}(1/\epsilon_g^{\frac{2r-1}{2r}})$

structure-exploiting method and presented an iteration complexity of $O(\max\{1/\epsilon_f^2, 1/\epsilon_g^2\})$ for it, when the upper-level objective is convex and the lower-level objective is convex and smooth. Note that imposing additional assumptions on the upper-level function, such as smoothness or strong convexity, does not result in faster rates for their method.

Recently, [JAMH23] presented a projection-free conditional gradient method (CG-BiO) that uses a cutting plane to approximate the solution set of the lower-level problem. Assuming both upper- and lower-level objective functions are convex and smooth, CG-BiO achieves a complexity of $\mathcal{O}(\max\{1/\epsilon_f, 1/\epsilon_g\})$. Since the suboptimality gap $f(\hat{\mathbf{x}}) - f^*$ may be negative for an infeasible point $\hat{\mathbf{x}}$, a more desirable metric is the *absolute suboptimality gap* $|f(\hat{\mathbf{x}}) - f^*|$. To ensure this, [JAMH23] introduced the Hölderian error bound condition on g. Specifically, under the r-th order Hölderian error bound condition, CG-BiO finds a solution $\hat{\mathbf{x}}$ with $|f(\hat{\mathbf{x}}) - f^*| \leq \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$ after $\mathcal{O}(\max\{1/\epsilon_f^r, 1/\epsilon_g\})$ iterations. More recently, [SBY23] introduced the regularized proximal accelerated method (R-APM), which runs the proximal accelerated gradient method on a weighted sum of the upper- and lower-level objective functions. Assuming both functions are convex and smooth, they established a complexity bound of $\mathcal{O}(\max\{1/\epsilon_f, 1/\epsilon_g\})$ to find an (ϵ_f, ϵ_g) solution. This bound is worse than the $\mathcal{O}(\max\{1/\sqrt{\epsilon_f}, 1/\epsilon_g\})$ complexity achieved by our proposed AGM-BiO method, assuming the feasible set $\mathcal{Z}$ is compact. Additionally, [SBY23] showed that when the lower-level objective function g satisfies the weak sharpness property (equivalent to the Hölderian error bound condition with $r = 1$), R-APM finds an (ϵ_f, ϵ_g)-absolute optimal solution after at most $\mathcal{O}(\max\{1/\sqrt{\epsilon_f}, 1/\sqrt{\epsilon_g}\})$ iterations. This result is comparable to our convergence result for AGM-BiO, which considers a more general Hölderian error bound condition.

Contribution. In this paper, we present a novel accelerated gradient-based bilevel optimization method with state-of-the-art non-asymptotic guarantees in terms of both suboptimality and infeasibility. At each iteration, our proposed AGM-BiO method uses a cutting plane to linearly approximate the solution set of the lower-level problem, and then runs a variant of the projected accelerated gradient update on the upper-level objective function. Next, we summarize our theo-

retical guarantees for the AGM-BiO method:

- When the feasible set $\mathcal{Z}$ is compact, we show that AGM-BiO finds $\hat{\mathbf{x}}$ that satisfies $f(\hat{\mathbf{x}}) - f^* \leq \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$ within $\mathcal{O}(\max\{1/\sqrt{\epsilon_f}, 1/\epsilon_g\})$ iterations, where f^* is the optimal value of problem (1) and g^* is the optimal value of the lower-level problem.
- With an additional r-th-order ($r \geq 1$) Hölderian error bound assumption on the lower-level problem, AGM-BiO finds $\hat{\mathbf{x}}$ satisfying $f(\hat{\mathbf{x}}) - f^* \leq \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$ within $\mathcal{O}(\max\{\epsilon_f^{-\frac{2r-1}{2r}}, \epsilon_g^{-\frac{2r-1}{2r}}\})$ iterations. Moreover, it achieves the stronger guarantee that $|f(\hat{\mathbf{x}}) - f^*| \leq \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$ within $\mathcal{O}(\max\{\epsilon_f^{-\frac{2r-1}{2}}, \epsilon_g^{-\frac{2r-1}{2r}}\})$ iterations.

These bounds all achieve the best-known complexity bounds in terms of both suboptimality and infeasibility guarantees for the considered settings. All the non-asymptotic results are summarized and compared in Table 1.

Discussions on two concurrent works. The authors in [WSJ24] proposed a bisection algorithm with a total operation complexity of $\tilde{\mathcal{O}}(\max\{\epsilon_f^{-0.5}, \epsilon_g^{-0.5}\})$ to find an (ϵ_f, ϵ_g)-optimal solution, assuming the upper-level objective f meets specific criteria. Specifically, Assumption 1(iv) in [WSJ24] implies the ability to compute the projection onto the sublevel set of the upper-level function f. However, this assumption may not hold for general functions, such as the mean squared loss function in our over-parameterized regression example in Section 5. In [CSJW24], the authors introduced the penalty-based accelerated proximal gradient method (PB-APG) for solving simple bilevel optimization problems with the r-th order Hölderian error bound assumption on the lower-level objective g. Their algorithm, similar to [SBY23], runs the accelerated proximal gradient method on a weighted sum of the upper and lower-level objective functions. PB-APG achieves a complexity of $\mathcal{O}(\epsilon_f^{-0.5r}) + \mathcal{O}(\epsilon_g^{-0.5})$ to find an (ϵ_f, ϵ_g)-optimal solution. The term $\mathcal{O}(1/\epsilon_f^{0.5r})$ can become significantly large as the order of the Hölderian error bound r increases. In contrast, our algorithm, AGM-BiO, avoids this issue, requiring at most $\tilde{\mathcal{O}}(\max\{\epsilon_f^{-\frac{2r-1}{2r}}, \epsilon_g^{-\frac{2r-1}{2r}}\})$ iterations to achieve an (ϵ_f, ϵ_g)-optimal solution. Therefore, regardless of how large r is, the worst-case complexity for AGM-BiO is $\tilde{\mathcal{O}}(\max\{\epsilon_f^{-1}, \epsilon_g^{-1}\})$. Thus, our method achieves a better rate than PB-APG when $r > 1$.

Additional related work. Previous work has explored "asymptotic" results for simple bilevel problems, dating back to Tikhonov-type regularization introduced in [TA77]. In this approach, the objectives of both levels are combined into a single-level problem using a regularization parameter $\sigma > 0$ and as $\sigma \to 0$ the solutions of the regularized single-level problem approaches a solution to the bilevel problem in (1). Further, Solodov [Sol07a] proposed the explicit descent method that solves problem (1) when upper and lower-level functions are smooth and convex. This result was further extended to a non-smooth setting in [Sol07b]. The results in both [Sol07a] and [Sol07b] only indicated that both upper and lower-level objective functions converge asymptotically. Moreover, Helou and Simões [HS17] proposed the ϵ-subgradient method to solve simple bilevel problems and showed its asymptotic convergence. Specifically, they assumed the upper-level objective function to be convex and utilized two different algorithms, namely, the Fast Iterative Bilevel Algorithm (FIBA) and Incremental Iterative Bilevel Algorithm (IIBA), that consider smooth and non-smooth lower-level objective functions, respectively.

Some studies have only established non-asymptotic convergence rates for the lower-level problem. One of the pioneering methods in this category is the minimal norm gradient (MNG) method, introduced by Beck and Sabach [BS14]. This method assumes that the upper-level objective function is smooth and strongly convex, while the lower-level objective function is smooth and convex. The

authors showed that the lower-level objective function reaches an iteration complexity of $\mathcal{O}(1/\epsilon^2)$. Subsequently, the Bilevel Gradient SAM (BiS-SAM) method was introduced by Sabach and Shtern [SS17], and it was proven to achieve a complexity of $\mathcal{O}(1/\epsilon)$ for the lower-level problem. A similar rate of convergence was also attained in [Mal17].

2 Preliminaries

In this section, we state the necessary assumptions and introduce the notions of optimality that we use in the paper.

2.1 Assumptions and Definitions

We focus on the case where both the upper and lower-level functions f and g are convex and smooth. Formally, we make the following assumptions.

Assumption 2.1. *Let $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^d$ and $\|\cdot\|_*$ be its dual norm. We assume these conditions hold:*

(i) $\mathcal{Z} \subset \mathbb{R}^d$ is convex and compact with diameter D, i.e., $\|\mathbf{x}-\mathbf{y}\| \leq D$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{Z}$.

(ii) g is convex and continuously differentiable on an open set containing $\mathcal{Z}$, and its gradient is L_g-Lipschitz, i.e., $\|\nabla g(\mathbf{x}) - \nabla g(\mathbf{y})\|_* \leq L_g\|\mathbf{x}-\mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{Z}$.

(iii) f is convex and continuously differentiable and its gradient is Lipschitz with constant L_f.

In this paper, we denote the optimal value and the optimal solution set of the lower-level problem as $g^* \triangleq \min_{\mathbf{z}\in\mathcal{Z}} g(\mathbf{z})$ and $\mathcal{X}_g^* \triangleq \operatorname{argmin}_{\mathbf{z}\in\mathcal{Z}} g(\mathbf{z})$, respectively. By Assumption 2.1, the set $\mathcal{X}_g^*$ is nonempty, compact, and convex, but in general, not a singleton since g could have multiple optimal solutions on $\mathcal{Z}$, as g is only convex but not strongly convex. Moreover, we use f^* to denote the optimal value and $\mathbf{x}^*$ to denote an optimal solution of problem (1).

Note that in the simple bilevel problem, the suboptimality of a solution $\hat{\mathbf{x}}$ is measured by the difference $f(\hat{\mathbf{x}}) - f^*$. Similarly, its infeasibility is indicated by $g(\hat{\mathbf{x}}) - g^*$. Since it is necessary to control both of these error terms to ensure minimal suboptimality and infeasibility, we formally define an (ϵ_f, ϵ_g)-optimal solution as follows.

Definition 2.1. *((ϵ_f, ϵ_g)-optimal solution). A point $\hat{\mathbf{x}} \in \mathcal{Z}$ is (ϵ_f, ϵ_g)-optimal for problem (1) if $f(\hat{\mathbf{x}}) - f^* \leq \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$.*

This definition is commonly used in simple bilevel optimization literature [JAMH23; MS23; CJAYM24; SBY23]. Due to the special structure of bilevel optimization, it is not guaranteed that $f(\hat{\mathbf{x}}) - f^*$ will always be positive. To address this, we propose the following definition, utilizing $|f(\hat{\mathbf{x}}) - f^*|$ as the absolute optimal criterion.

Definition 2.2. *((ϵ_f, ϵ_g)-absolute optimal solution). A point $\hat{\mathbf{x}} \in \mathcal{Z}$ is (ϵ_f, ϵ_g)-absolute optimal for problem (1) if $|f(\hat{\mathbf{x}}) - f^*| \leq \epsilon_f$ and $|g(\hat{\mathbf{x}}) - g^*| \leq \epsilon_g$.*

3 Algorithm

Before presenting our method, we first introduce a conceptual accelerated gradient method for solving the simple bilevel problem in (1). The first step is to recast it as a constrained optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; f(\mathbf{x}) \quad \text{s.t.} \quad \mathbf{x} \in \mathcal{X}_g^*, \tag{2}$$

where $\mathcal{X}_g^* \triangleq \operatorname{argmin}_{\mathbf{z}\in\mathcal{Z}} \; g(\mathbf{z})$ denotes the solution set of the lower-level objective. Thus, conceptually, we can apply Nesterov's accelerated gradient method (AGM) to obtain a rate of $O(1/k^2)$ on the upper-level objective f. Several variants of AGM have been proposed in the literature; see, e.g., [dST+21]. In this paper, we consider a variant proposed in [Tse08]. It involves three intertwined sequences of iterates $\{\mathbf{x}_k\}_{k\geq 0}, \{\mathbf{y}_k\}_{k\geq 0}, \{\mathbf{z}_k\}_{k\geq 0}$ and the scalar variables $\{a_k\}_{k\geq 0}$ and $\{A_k\}_{k\geq 0}$. In the first step, we compute the auxiliary iterate $\mathbf{y}_k$ by

$$\mathbf{y}_k = \frac{A_k}{A_k + a_k}\mathbf{x}_k + \frac{a_k}{A_k + a_k}\mathbf{z}_k. \tag{3}$$

Then in the second step, we update $\mathbf{z}_{k+1}$ by

$$\mathbf{z}_{k+1} = \Pi_{\mathcal{X}_g^*}(\mathbf{z}_k - a_k \nabla f(\mathbf{y}_k)), \tag{4}$$

where $\Pi_{\mathcal{X}_g^*}(\cdot)$ denotes the Euclidean projection onto the set $\mathcal{X}_g^*$. Finally, in the third step, we compute $\mathbf{x}_{k+1}$ and A_{k+1}

$$\mathbf{x}_{k+1} = \frac{A_k}{A_k + a_k}\mathbf{x}_k + \frac{a_k}{A_k + a_k}\mathbf{z}_{k+1} \tag{5}$$

and

$$A_{k+1} = A_k + a_k \tag{6}$$

It can be shown that if the stepsize is selected as $a_k = \frac{k+1}{4L_f}$, then the suboptimality gap $f(\mathbf{x}_k) - f(\mathbf{x}^*)$ of the iterates generated by the method above converges to zero at the optimal rate of $\mathcal{O}(1/k^2)$. In this case, indeed all the iterates are feasible as it is possible to project onto the set $\mathcal{X}_g^*$. However, the conceptual method above is not directly implementable for the simple bilevel problem considered in this paper, as the constraint set $\mathcal{X}_g^*$ is not explicitly given. As a result projection onto the set $\mathcal{X}_g^*$ is not computationally tractable.

To address this issue, our key idea involves replacing the implicit set $\mathcal{X}_g^*$ in (4) with another set, $\mathcal{X}_k$, which can be explicitly characterized. This approach makes it feasible to perform the Euclidean projection onto $\mathcal{X}_k$. Additionally, $\mathcal{X}_k$ must possess another important property: it should encompass the optimal solution set $\mathcal{X}_g^*$.

Inspired by the cutting plane approach in [JAMH23], we let the set $\mathcal{X}_k$ be the intersection of $\mathcal{Z}$ and a halfspace:

$$\mathcal{X}_k \triangleq \{\mathbf{z} \in \mathcal{Z} : g(\mathbf{y}_k) + \langle \nabla g(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle \leq g_k\}. \tag{7}$$

Here, the auxiliary sequence $\{g_k\}_{k\geq 0}$ should be selected such that $g_k \geq g^*$ and $g_k \to g^*$. One straightforward way to generate this sequence is by applying an accelerated projected gradient method to the lower-level objective g separately. The loss function of the iterates generated by this algorithm can be considered as $\{g_k\}$ for the above halfspace. Note that in this case, it is known that

$$0 \leq g_k - g^* \leq \frac{2L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{(k+1)^2}, \quad \forall k \geq 0. \tag{8}$$

Algorithm 1 Accelerated Gradient Method for Bilevel Optimization (AGM-BiO)

Input: A sequence $\{g_k\}_{k=0}^K$, a scalar $\gamma \in (0, 1]$
Initialization: $A_0 = 0$, $\mathbf{x}_0 = \mathbf{z}_0 \in \mathbb{R}^n$
for $k = 0, \ldots, K$ **do**
  Set $a_k = \gamma \dfrac{k+1}{4L_f}$
  Compute $\mathbf{y}_k = \dfrac{A_k}{A_k + a_k}\mathbf{x}_k + \dfrac{a_k}{A_k + a_k}\mathbf{z}_k$
  Compute $\mathbf{z}_{k+1} = \Pi_{\mathcal{X}_k}(\mathbf{z}_k - a_k \nabla f(\mathbf{y}_k))$, where

$$\mathcal{X}_k \triangleq \{\mathbf{z} \in \mathcal{Z} : g(\mathbf{y}_k) + \langle \nabla g(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle \leq g_k\}$$

  Compute $\mathbf{x}_{k+1} = \dfrac{A_k}{A_k + a_k}\mathbf{x}_k + \dfrac{a_k}{A_k + a_k}\mathbf{z}_{k+1}$
  Update $A_{k+1} = A_k + a_k$
end for
Return: $\mathbf{x}_K$

Hence, the above requirements on the sequence $\{g_k\}$ are satisfied. Two remarks on the set $\mathcal{X}_k$ are in order. First, the set $\mathcal{X}_k$ in (7) has an explicit form, and thus computing the Euclidean projection onto $\mathcal{X}_k$ is tractable. It also can be verified that the set $\mathcal{X}_k$ always contains the lower-level problem solution set $\mathcal{X}_g^*$. To prove this, let $\hat{\mathbf{x}}^*$ be any point in $\mathcal{X}_g^*$. By using the convexity of g, we obtain that $g(\mathbf{y}_k) + \langle \nabla g(\mathbf{y}_k), \hat{\mathbf{x}}^* - \mathbf{y}_k \rangle \leq g(\hat{\mathbf{x}}^*) = g^* \leq g_k$. Thus $\hat{\mathbf{x}}^*$ indeed satisfies both constraints in (7) and we obtain that $\hat{\mathbf{x}}^* \in \mathcal{X}_k$.

Now that we have identified an appropriate replacement for the set $\mathcal{X}_g^*$, we can easily implement a variant of the projected accelerated gradient method for the bilevel problem using the surrogate set $\mathcal{X}_k$. We refer to our method as the Accelerated Gradient Method for Bilevel Optimization (AGM-BiO) and its steps are outlined in Algorithm 1. It is important to note that the iterates, when projected onto the set $\mathcal{X}_k$, may not belong to the set $\mathcal{X}_g^*$, as $\mathcal{X}_k$ is an approximation of the true solution set. Consequently, the iterates might be infeasible. However, the design of $\mathcal{X}_k$ allows us to control the infeasibility of the iterates, as we will demonstrate in the convergence analysis section.

Remark 3.1. *The design of the halfspace as specified in (7) should be recognized as a nuanced task. Various alternative formulations of halfspaces could fulfill the same primary conditions, such as* $\{\mathbf{z} \in \mathbb{R}^d : g(\mathbf{x}_k) + \langle \nabla g(\mathbf{x}_k), \mathbf{z} - \mathbf{x}_k \rangle \leq g_k\}$ *and* $\{\mathbf{z} \in \mathbb{R}^d : g(\mathbf{z}_k) + \langle \nabla g(\mathbf{z}_k), \mathbf{z} - \mathbf{z}_k \rangle \leq g_k\}$. *However, the selection of the gradient at* $\mathbf{y}_k$ *for constructing the halfspace is not arbitrary but essential as we characterize in the convergence analysis of our method.*

Remark 3.2. *While our paper focuses on the smooth setting, our algorithm can be extended to the composite setting with an additional assumption and some proof modifications. Due to limited space, please refer to Section B in the Appendix for more details.*

4 Convergence Analysis

In this section, we analyze the convergence rate and iteration complexity of our proposed AGM-BiO method for convex simple bilevel optimization problems. We choose the stepsize $a_k = (k+1)/(4L_f)$,

which is inspired from our theoretical analysis. The main theorem is as follows,

Theorem 4.1. *Suppose Assumption 2.1 holds. Let $\{\mathbf{x}_k\}_{k\geq 0}$ be the sequence of iterates generated by Algorithm 1 with stepsize $a_k = (k+1)/(4L_f)$ for $k \geq 0$ and suppose the sequence g_k used for generating the cutting plane satisfies (8). Then, for any $k \geq 0$ we have,*

(i) The function suboptimality is bounded above by

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{4L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{k(k+1)} \tag{9}$$

(ii) The infeasibility term is bounded above by

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)} + \frac{2L_gD^2}{k+1} \tag{10}$$

(iii) Furthermore, if the condition $f(\mathbf{x}_k) \geq f(\mathbf{x}^*)$ holds, then the infeasibility term is bounded above by

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{8L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)}. \tag{11}$$

Theorem 4.1 shows the upper-level objective function gap is upper bounded by $\mathcal{O}(1/k^2)$, which matches the convergence rate of the accelerated gradient method for single-level optimization problems. On the other hand, the suboptimality of the lower-level objective which measures infeasibility for the bilevel problem in the worst case is bounded above by $\mathcal{O}(1/k)$. In cases where the $f(\mathbf{x}_k) \geq f^*$ this upper bound could be even improved to $\mathcal{O}(1/k^2)$. As a corollary of the worst-case bounds, Algorithm 1 will return an (ϵ_f, ϵ_g)-optimal solution after at most the following number of iterations $\mathcal{O}\left(\max\left\{\frac{1}{\sqrt{\epsilon_f}}, \frac{1}{\epsilon_g}\right\}\right)$. We should emphasize that, under the assumptions being considered, this complexity bound represents the best-known bound among all previous works summarized in Table 1. It is worth noting that concurrent work by [SBY23] achieves $\mathcal{O}(\max\{\frac{1}{\epsilon_f}, \frac{1}{\epsilon_g}\})$ under similar assumptions. They can only improve their complexity bound with the additional assumption that g satisfies the weak sharpness condition.

Remark 4.1 (The necessity of compactness of $\mathcal{Z}$)**.** *For the lower-level objective, we show that $A_k(g(\mathbf{x}_k) - g(\mathbf{x}^*)) \leq \sum_{i=0}^{k-1} a_i(g_i - g^*) + \frac{L_g}{4L_f}\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2$ ((33) in Section 4). The main challenge in obtaining an accelerated rate of $\mathcal{O}(1/k^2)$ for g is controlling $\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2$. Without a lower bound on f, this term cannot be bounded by the upper-level suboptimality alone. If $f(\mathbf{x}_k) \geq f(\mathbf{x}^*)$, we can achieve the rate of $\mathcal{O}(1/k^2)$ for g. Otherwise, we use the compactness of $\mathcal{Z}$ to achieve $\mathcal{O}(1/k)$ for g. Please refer to Section 4 for more details.*

Remark 4.2 (Removable log terms)**.** *The* log *terms in all the complexity results can be removed by choosing the auxiliary sequence $g_k = g_K$ for all $0 \leq k \leq K$, which satisfies the condition (8). This eliminates the* log *term in (33) and all subsequent results. However, this choice of $\{g_k\}_{k\geq 0}$ requires predetermining the total number of iterations K.*

Since the algorithm's output $\hat{\mathbf{x}}$ may fall outside the feasible set $\mathcal{X}_g^*$, the expression $f(\hat{\mathbf{x}}) - f^*$ may not necessarily be non-negative. On the other hand, under the considered assumptions, proving convergence in terms of $|f(\hat{\mathbf{x}}) - f^*|$ is known to be impossible due to a negative result presented by [CXZ23]. Specifically, for any first-order method and a given number of iterations k, they demonstrated the existence of an instance of Problem (1) where $|f(\mathbf{x}_k) - f^*| \geq 1$ for all $k \geq 0$. Thus, to provide any form of guarantee in terms of the absolute value of the suboptimality, i.e., $|f(\hat{\mathbf{x}}) - f^*|$, we need an additional assumption to obtain a lower bound on suboptimality and to provide a convergence bound for $|f(\hat{\mathbf{x}}) - f^*|$. We will address this point in the following section.

4.1 Convergence under Hölderian Error Bound

In this section, we introduce an additional regularity condition on g to establish a lower bound for $f(\hat{\mathbf{x}}) - f^*$. Formally, we assume that the lower-level objective function, g, satisfies the Hölderian Error Bound condition. This condition governs how the objective value $g(\mathbf{x})$ grows as the point $\mathbf{x}$ moves away from the optimal solution set $\mathcal{X}_g^*$. Intuitively, since our method's output $\hat{\mathbf{x}}$ is ϵ_g-optimal for the lower-level problem, it should be close to the optimal solution set $\mathcal{X}_g^*$ when this regularity condition on g is met. Consequently, we can utilize this proximity to establish a lower bound for $f(\hat{\mathbf{x}}) - f^*$ by leveraging the smoothness property of f.

Assumption 4.1. *The function g satisfies the Hölderian error bound for some $\alpha > 0$ and $r \geq 1$, i.e,*

$$\frac{\alpha}{r} \operatorname{dist}\left(\mathbf{x}, \mathcal{X}_g^*\right)^r \leq g(\mathbf{x}) - g^*, \quad \forall \mathbf{x} \in \mathcal{Z}, \tag{12}$$

where $\operatorname{dist}\left(\mathbf{x}, \mathcal{X}_g^*\right) \triangleq \inf_{\mathbf{x}' \in \mathcal{X}_g^*} \|\mathbf{x} - \mathbf{x}'\|$.

We note that the Hölderian error bound condition in (12) is well-studied in the optimization literature [Pan97; BNPS17; Rd17] and is known to hold in general when function g is analytic and the set $\mathcal{Z}$ is bounded [LP94]. There are two important special cases of the Hölderian error bound condition: 1) g satisfies (12) with $r = 1$ known as the weak sharpness condition [BF93; BD05]; 2) g satisfies (12) with $r = 2$ known as the quadratic functional growth condition [DL18].

By using the Hölderian error bound condition, Jiang, Abolfazli, Mokhtari, and Hamedani [JAMH23] established a stronger relation between suboptimality and infeasibility, as shown in the following proposition.

Proposition 4.2 ([JAMH23, Proposition 1])**.** *Assume that f is convex and g satisfies Assumption 4.1, and define $M = \max_{\mathbf{x} \in \mathcal{X}_g^*} \|\nabla f(\mathbf{x})\|_*$. Then it holds that $f(\hat{\mathbf{x}}) - f^* \geq -M(\frac{r(g(\hat{\mathbf{x}}) - g^*)}{\alpha})^{\frac{1}{r}}$ for any $\hat{\mathbf{x}} \in \mathcal{Z}$.*

Hence, under Assumption 4.1, Proposition 4.2 shows that the suboptimality $f(\hat{\mathbf{x}}) - f^*$ can also be bounded from below when $\hat{\mathbf{x}}$ is an approximate solution of the lower-level problem. As a result, we can establish a convergence bound on $|f(\mathbf{x}_k) - f^*|$ by combining Proposition 4.2 with the upper bounds in Theorem 4.1. Moreover, it also allows us to improve the convergence rate for the lower-level problem. To prove this claim, we first introduce the following lemma which establishes an upper bound on the weighted sum of upper and lower-level objectives.

Lemma 4.3. *Suppose conditions (ii) and (iii) in Assumption 2.1 hold. Let $\{\mathbf{x}_k\}$ be the sequence of iterates generated by Algorithm 1 with stepsize $a_k = \gamma(k+1)/(4L_f)$, where $0 < \gamma \leq 1$. Moreover, suppose the sequence g_k used for generating the cutting plane satisfies satisfies (8). Then, for any $\lambda \geq \frac{L_g}{(2/\gamma - 1)L_f}$ and $k \geq 0$ we have*

$$\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)} + \frac{4\lambda L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)} \tag{13}$$

This result characterizes and upper bound of $\mathcal{O}(1/k^2)$ on the expression $\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + g(\mathbf{x}_k) - g(\mathbf{x}^*)$. That said, the first term in this expression, a.k.a., $\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*))$ may not be non-negative for a bilevel problem as discussed earlier. Hence, we cannot simply eliminate $\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*))$ to show an upper bound of $\mathcal{O}(1/k^2)$ on infeasibility, a.k.a., $g(\mathbf{x}_k) - g(\mathbf{x}^*)$. Instead, we leverage the Hölderian error bound on g and apply Proposition 4.2 to the first term. As a result, we can eliminate the dependence on f in (13). In this case, we can establish an upper bound on infeasibility.

Theorem 4.4. *Suppose conditions (ii) and (iii) in Assumption 2.1 hold and the lower-level function g satisfies the Hölderian error bound with $r>1$. Let $\{\mathbf{x}_k\}$ be the iterates generated by Algorithm 1 with stepsize $a_k = \gamma\frac{k+1}{4L_f}$, where $\gamma = 1/(\frac{2L_g}{L_f}T^{\frac{2r-2}{2r-1}}+2)$ and T is the total number of iterations that we run the algorithm. Moreover, suppose the sequence g_k used for generating the cutting plane satisfies (8). If we define the constants $C_f \triangleq 8L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2$, $C_g \triangleq 12L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2$ and $C \triangleq M(\frac{r}{\alpha})^{\frac{1}{r}}$, where $M \triangleq \max_{\mathbf{x}\in\mathcal{X}_g^*}\|\nabla f(\mathbf{x})\|$, α and r are the parameters in Assumption 4.1, then the following results hold:*

(i) The function suboptimality is bounded above by

$$f(\mathbf{x}_T)-f(\mathbf{x}^*) \leq \frac{C_g(\ln T+1)}{T^{2r/(2r-1)}}+\frac{C_f}{T^2} \tag{14}$$

(ii) The function suboptimality is bounded below by

$$f(\mathbf{x}_T)-f(\mathbf{x}^*) \geq -C\max\left\{\frac{(2C_g(\ln T+1))^{\frac{1}{r}}}{T^{\frac{2}{2r-1}}}+\frac{(2C_f)^{\frac{1}{r}}}{T^{\frac{2}{r}}},\frac{(2C)^{\frac{1}{r-1}}}{T^{\frac{2}{2r-1}}}\right\}$$

(iii) The infeasibility term is bounded above by

$$g(\mathbf{x}_T)-g(\mathbf{x}^*) \leq \max\left\{\frac{2C_g(\ln T+1)}{T^{2r/(2r-1)}}+\frac{2C_f}{T^2},\frac{(2C)^{\frac{r}{r-1}}}{T^{\frac{2r}{2r-1}}}\right\} \tag{15}$$

Before unfolding this result, we would like to highlight that unlike the result in Theorem 4.1, the above bounds in Theorem 4.4 do not require the feasible set to be compact. Since $r>1$, the first result shows $f(\mathbf{x}_T)-f(\mathbf{x}^*)$ has an upper bound of $\mathcal{O}((\frac{1}{T})^{\frac{2r}{2r-1}})$ and the second result guarantees a lower bound of $-\mathcal{O}((\frac{1}{T})^{\frac{2}{2r-1}})$. These two bounds together lead to an upper bound of $\mathcal{O}((\frac{1}{T})^{\frac{2}{2r-1}})$ for the absolute error $|f(\mathbf{x}_T)-f(\mathbf{x}^*)|$. Moreover, the third result implies that the lower-level problem suboptimality which measures infeasibility is bounded above by $\mathcal{O}((\frac{1}{T})^{\frac{2r}{2r-1}})$.

The previous result presented in Theorem 4.4 is applicable when $r>1$. However, for the case that 1st-order Hölderian error bound condition on g holds (i.e., weak sharpness condition), we require a distinct analysis and a different choice of γ to achieve the tightest bounds. In the subsequent theorem, we present our findings for this specific scenario.

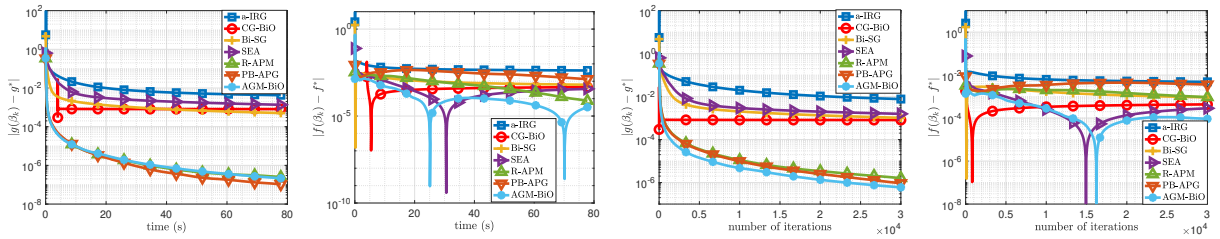
Theorem 4.5. *Suppose conditions (ii) and (iii) in Assumption 2.1 are met and that the lower-level objective function g satisfies the Hölderian error bound with $r=1$. Let $\{\mathbf{x}_k\}$ be the sequence of iterates generated by Algorithm 1 with stepsize $a_k = \gamma\frac{k+1}{4L_f}$, where $0<\gamma\leq\min\{\frac{2\alpha L_f}{2ML_g+\alpha L_f},1\}$. Moreover, suppose the sequence g_k used for generating the cutting plane satisfies (8), and recall $M \triangleq \max_{\mathbf{x}\in\mathcal{X}_g^*}\|\nabla f(\mathbf{x})\|$ and α in Assumption 4.1. If we define the constants $C_f \triangleq 4L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2$ and $C_g \triangleq 8L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2$, then for any $k\geq 0$:*

(i) The function suboptimality is bounded above by

$$f(\mathbf{x}_k)-f(\mathbf{x}^*) \leq \frac{C_f}{\gamma k(k+1)}. \tag{16}$$

(ii) The function suboptimality is bounded below by

$$f(\mathbf{x}_k)-f(\mathbf{x}^*) \geq -\frac{C_gM(\ln k+1)}{\alpha k(k+1)}-\frac{C_f}{\gamma k(k+1)}. \tag{17}$$



(a) Infeas. vs. time (b) Subopt. vs. time (c) Infeas. vs. iterations (d) Subopt. vs. iterations

Figure 1: Comparison of a-IRG, CG-BiO, Bi-SG, SEA, R-APM, PB-APG, and AGM-BiO for solving the over-parameterized regression problem.

(iii) The infeasibility term is bounded above by

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{C_g(\ln k + 1)}{k(k+1)} + \frac{\alpha C_f}{\gamma M k(k+1)}. \tag{18}$$

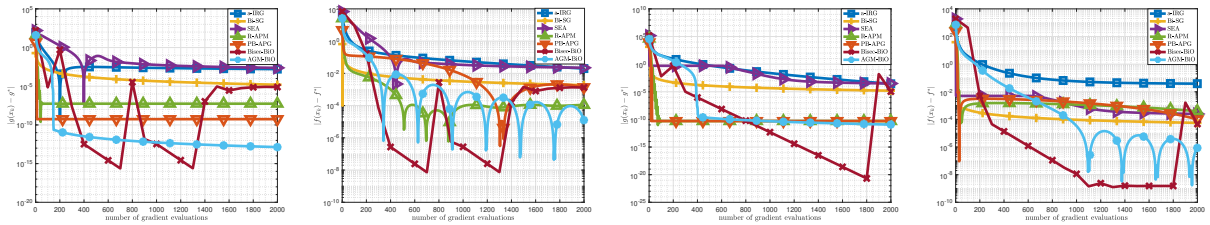
Theorem 4.5 shows that under the Hölderian error bound with $r = 1$, also known as weak sharpness condition, the absolute value of the function suboptimality $|f(\mathbf{x}_k) - f(\mathbf{x}^*)|$ approaches zero at a rate of $\mathcal{O}(1/k^2)$ – ignoring the log term. The lower-level error $g(\mathbf{x}_k) - g(\mathbf{x}^*)$, capturing the infeasibility of the iterates, also approaches zero at a rate of $\mathcal{O}(1/k^2)$. As a corollary, Algorithm 1 returns an (ϵ_f, ϵ_g)-absolute optimal solution after $\mathcal{O}(\max\{\frac{1}{\sqrt{\epsilon_f}}, \frac{1}{\sqrt{\epsilon_g}}\})$ iterations. This iteration complexity matches the result in a concurrent work [SBY23] under similar assumptions.

5 Numerical Experiments

In this section, we evaluate our AGM-BiO method on two different bilevel problems using real and synthetic datasets. We compare its runtime and iteration count with other methods, including a-IRG [KY21], CG-BiO [JAMH23], Bi-SG [MS23], SEA [SHK23], R-APM [SBY23], PB-APG [CSJW24], and Bisec-BiO [WSJ24].

Over-parameterized regression. We examine problem (1) where the lower-level problem corresponds to training loss, and the upper-level pertains to validation loss. The objective is to minimize the validation loss by selecting an optimal training loss solution. This method is also referred to as lexicographic optimization [GL21]. A common example of that is the constrained regression problem, where we aim to find an optimal parameter vector $\boldsymbol{\beta} \in \mathbb{R}^d$ for the validation loss that minimizes the loss $\ell_{\text{tr}}(\boldsymbol{\beta})$ over the training dataset $\mathcal{D}_{\text{tr}}$. To represent some prior knowledge, we constrain $\boldsymbol{\beta}$ to be in some subset $\mathcal{Z} \subseteq \mathbb{R}^d$, e.g., $\mathcal{Z} = \{\boldsymbol{\beta} \mid \beta_1 \leq \cdots \leq \beta_d\}$ in isotonic regression and $\mathcal{Z} = \{\boldsymbol{\beta} \mid \|\boldsymbol{\beta}\|_p \leq \lambda\}$ in L_p constrained regression. Without explicit regularization, an over-parameterized regression over the training dataset has multiple global minima, but not all these optimal regression coefficients perform equally on validation or testing datasets. Thus, the upper-level objective serves as a secondary criterion to ensure a smaller error on the validation dataset $\mathcal{D}_{\text{val}}$. The problem can be cast as

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} f(\boldsymbol{\beta}) \triangleq \ell_{\text{val}}(\boldsymbol{\beta}) \quad \text{s.t.} \quad \boldsymbol{\beta} \in \operatorname*{argmin}_{\mathbf{z} \in \mathcal{Z}} g(\mathbf{z}) \triangleq \ell_{\text{tr}}(\mathbf{z})$$



(a) Infeas. ($n = 3$) (b) Subopt. ($n = 3$) (c) Infeas. ($n = 100$) (d) Subopt. ($n = 100$)

Figure 2: Comparison of a-IRG, Bi-SG, SEA, R-APM, PB-APG, Bisec-BiO, and AGM-BiO for solving the linear inverse problem.

In this case, both upper-level and lower-level objectives are convex and smooth if the loss ℓ is smooth and convex. Since projections onto the sublevel set of f are difficult to compute, Bisec-BiO is excluded from this experiment.

We apply the Wikipedia Math Essential dataset [Roz+21] which is composed of a data matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$ with $n = 1068$ samples and $d = 730$ features and an output vector $\mathbf{b} \in \mathbb{R}^n$. We use 75% of the dataset as the training set $(\mathbf{A}_{\text{tr}}, \mathbf{b}_{\text{tr}})$ and 25% as the validation set $(\mathbf{A}_{\text{val}}, \mathbf{b}_{\text{val}})$. For both upper- and lower-level loss functions, we use the least squared loss. Then the lower-level objective is $g(\boldsymbol{\beta}) = \frac{1}{2}\|\mathbf{A}_{tr}\boldsymbol{\beta} - \mathbf{b}_{tr}\|_2^2$, the upper-level objective is $f(\boldsymbol{\beta}) = \frac{1}{2}\|\mathbf{A}_{val}\boldsymbol{\beta} - \mathbf{b}_{val}\|_2^2$, and the constraint set is chosen as the unit L_2-ball $\mathcal{Z} = \{\boldsymbol{\beta} \mid \|\boldsymbol{\beta}\|_2 \leq 1\}$. Note that this regression problem is over-parameterized since the number of features d is larger than the number of data points in both the training set and validation set.

In Figures 1(a) and 1(c), we observe that the three accelerated gradient-based methods (R-APM, PB-APG, and AGM-BiO) converge faster in reducing infeasibility, both in terms of runtime and number of iterations. In terms of absolute suboptimality, shown in Figures 1(b) and 1(d), AGM-BiO achieves the smallest absolute suboptimality gap among all algorithms. Unlike the infeasibility plots, R-APM and PB-APG underperform compared to AGM-BiO. Note that the lower-level objective in this problem does not satisfy the weak sharpness condition, so the regularization parameter η in R-APM is set as $1/(K+1)$. Consequently, the suboptimality for R-APM converges slower than AGM-BiO, as suggested by the theoretical results in Table 1.

Linear inverse problems. In the next experiment, we concentrate on a problem that fulfills the Hölderian Error Bound condition for some $r > 1$. We aim to evaluate the performance of our method in this specific context and verify the validity of our theoretical results for this scenario. Specifically, we focus on the so-called linear inverse problems, commonly used to evaluate convex bilevel optimization algorithms, which originate from [SS17]. The goal of linear inverse problems is to obtain a solution $\mathbf{x} \in \mathbb{R}^n$ to the system of linear equation $\mathbf{A}\mathbf{x} = \mathbf{b}$. Note that if $\mathbf{A}$ is rank-deficient, there can be multiple solutions, or there might be no exact solution due to noise. To address this issue, we chase a solution that has the smallest weighted norm with respect to some positive definite matrix $\mathbf{Q}$, i.e., $\|\mathbf{x}\|_{\mathbf{Q}} := \sqrt{\mathbf{x}^\top \mathbf{Q}\mathbf{x}}$. This problem can be also cast as the following simple bilevel problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \triangleq \frac{1}{2}\|\mathbf{x}\|_{\mathbf{Q}}^2 \quad \text{s.t.} \quad \mathbf{x} \in \operatorname*{argmin}_{\mathbf{z}\in\mathcal{Z}} g(\mathbf{z}) \triangleq \frac{1}{2}\|\mathbf{A}\mathbf{z} - \mathbf{b}\|_2^2$$

For this class of problem, if $\mathbf{Q}$, $\mathbf{A}$, and $\mathbf{b}$ are generated randomly or by the "regularization tools" like [SS17; SHK23], we are not able to obtain the exact optimal value f^*. To the best of our

knowledge, no existing solver could obtain the exact optimal value f^* for this bilevel problem. Specifically, the existing solvers either fail to solve this bilevel problem or return an inaccurate solution by solving a relaxed version of the problem. Hence, in [SS17; SHK23] they only reported the upper-level function value. However, in this paper, we intend to obtain the complexity bounds for finding (ϵ_f, ϵ_g)-optimal and (ϵ_f, ϵ_g)-absolute optimal solutions. Without knowing f^*, we can not characterize the behavior of $|f(\mathbf{x}_k) - f^*|$. Therefore, we choose an example where we can obtain the exact solution. Specifically, we set $\mathbf{Q} = \mathbf{I}_n$, $\mathbf{A} = \mathbf{1}_n^\top$, $\mathbf{b} = 1$, and the constraint set $\mathcal{Z} = \mathbb{R}^n_+$. In this case, the optimal solution $\mathbf{x}^* = \frac{1}{n}\mathbf{1}_n$ and optimal value $f^* = \frac{1}{2n}$. This specific example essentially involves seeking the minimum norm for an under-determined system. Note that the lower-level objective in this problem satisfies the Hölderian Error Bound condition with order $r = 2$ [NNG19]. Hence, we do not need the constraint set $\mathcal{Z}$ to be compact as shown in Theorem 4.4. Due to the unbounded nature of the constraint set, Frank-Wolfe-type methods are not viable options. Consequently, we have opted not to incorporate CG-BiO in this experiment.

We explored examples with two distinct dimensions: $n = 3$ and $n = 100$, evaluating a total of 2000 gradients. In Figures 2(a) and 2(c), AGM-BiO shows superior performance in terms of infeasibility. In Figures 2(b) and 2(d), we compare methods in terms of absolute error of suboptimality. The gap between R-APM and AGM-BiO is smaller for $n = 3$, but for $n = 100$, AGM-BiO significantly outperforms all other methods, including R-APM. Since the regularization and penalty parameters in R-APM and PB-APG are fixed, they might get stuck at a certain accuracy level, as seen in Figures 2(a) and 2(c). In contrast, AGM-BiO uses a dynamic framework for minimizing the upper and lower-level functions, consistently reducing both suboptimality and infeasibility. Although Bisec-BiO theoretically has the best complexity results due to the ease of projecting onto the sublevel set of f, its performance in the last iteration is inconsistent, as shown in Figure 2.

6 Conclusion

In this paper, we introduced an accelerated gradient-based algorithm for solving a specific class of bilevel optimization problems with convex objective functions in both the upper and lower levels. Our proposed algorithm achieves a computational complexity of $\mathcal{O}(\max\{\epsilon_f^{-0.5}, \epsilon_g^{-1}\})$. When an additional weak sharpness condition is applied to the lower-level function g, the iteration complexity improves to $\tilde{\mathcal{O}}(\max\{\epsilon_f^{-0.5}, \epsilon_g^{-0.5}\})$, matching the well-known fastest convergence rate for single-level convex optimization problems. We further extended this result to an iteration complexity of $\tilde{\mathcal{O}}(\max\{\epsilon_f^{-\frac{2r-1}{2r}}, \epsilon_g^{-\frac{2r-1}{2r}}\})$ when the lower-level loss satisfies the Hölderian error bound assumption.

Acknowledgements

The research of J. Cao, R. Jiang and A. Mokhtari is supported in part by NSF Grants 2127697, 2019844, and 2112471, ARO Grant W911NF2110226, the Machine Learning Lab (MLL) at UT Austin, and the Wireless Networking and Communications Group (WNCG) Industrial Affiliates Program. The research of E. Yazdandoost Hamedani is supported by NSF Grant 2127696.

References

[ABC19] Samir Adly, Loic Bourdin, and Fabien Caubet. "On a decomposition formula for the proximal operator of the sum of two convex functions". In: *Journal of Convex Analysis* 26.2 (2019), pp. 699–718 (page 24).

[BBW18] Heinz H Bauschke, Minh N Bui, and Xianfu Wang. "Projecting onto the intersection of a cone and a sphere". In: *SIAM Journal on Optimization* 28.3 (2018), pp. 2158–2188 (page 24).

[Bau+11] Heinz H Bauschke, Regina S Burachik, Patrick L Combettes, Veit Elser, D Russell Luke, and Henry Wolkowicz. *Fixed-point algorithms for inverse problems in science and engineering*. Vol. 49. Springer Science & Business Media, 2011 (page 29).

[BS14] Amir Beck and Shoham Sabach. "A first order method for finding minimal norm-like solutions of convex optimization problems". In: *Mathematical Programming* 147.1-2 (2014), pp. 25–46 (page 4).

[BT09] Amir Beck and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In: *SIAM journal on imaging sciences* 2.1 (2009), pp. 183–202 (pages 2, 24).

[BHTV18] Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. "Meta-learning with differentiable closed-form solvers". In: *arXiv preprint arXiv:1805.08136* (2018) (page 2).

[BNPS17] Jérôme Bolte, Trong Phong Nguyen, Juan Peypouquet, and Bruce W Suter. "From error bounds to the complexity of first-order descent methods for convex functions". In: *Mathematical Programming* 165 (2017), pp. 471–507 (page 9).

[BMK20] Zalán Borsos, Mojmir Mutny, and Andreas Krause. "Coresets via bilevel optimization for continual learning and streaming". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 14879–14890 (page 2).

[BD05] James V Burke and Sien Deng. "Weak sharp minima revisited, part II: application to linear regularity and error bounds". In: *Mathematical programming* 104 (2005), pp. 235–261 (page 9).

[BF93] James V Burke and Michael C Ferris. "Weak sharp minima in mathematical programming". In: *SIAM Journal on Control and Optimization* 31.5 (1993), pp. 1340–1359 (page 9).

[CJAYM24] Jincheng Cao, Ruichen Jiang, Nazanin Abolfazli, Erfan Yazdandoost Hamedani, and Aryan Mokhtari. "Projection-free methods for stochastic simple bilevel optimization with convex lower-level problem". In: *Advances in Neural Information Processing Systems* 36 (2024) (pages 2, 5).

[CXZ23] Lesi Chen, Jing Xu, and Jingzhao Zhang. "On Bilevel Optimization without Lower-level Strong Convexity". In: *arXiv preprint arXiv:2301.00712* (2023) (page 8).

[CSJW24] Pengyu Chen, Xu Shi, Rujun Jiang, and Jiulin Wang. "Penalty-based Methods for Simple Bilevel Optimization under H\"{o} lderian Error Bounds". In: *arXiv preprint arXiv:2402.02155* (2024) (pages 3, 4, 11, 24).

[dST+21] Alexandre d'Aspremont, Damien Scieur, Adrien Taylor, et al. "Acceleration methods". In: *Foundations and Trends® in Optimization* 5.1-2 (2021), pp. 1–245 (page 6).

[DDD10] Stephen Dempe, Nguyen Dinh, and Joydeep Dutta. "Optimality conditions for a simple convex bilevel programming problem". In: *Variational Analysis and Generalized Differentiation in Optimization and Control: In Honor of Boris S. Mordukhovich* (2010), pp. 149–161 (page 2).

[DL18] Dmitriy Drusvyatskiy and Adrian S Lewis. "Error bounds, quadratic growth, and linear convergence of proximal methods". In: *Mathematics of Operations Research* 43.3 (2018), pp. 919–948 (page 9).

[DP20] Joydeep Dutta and Tanushree Pandit. "Algorithms for simple bilevel programming". In: *Bilevel Optimization: Advances and Next Challenges* (2020), pp. 253–291 (page 2).

[FFSGP18] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. "Bilevel programming for hyperparameter optimization and meta-learning". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 1568–1577 (page 2).

[GL21] Chengyue Gong and Xingchao Liu. "Bi-objective trade-off with dynamic barrier gradient descent". In: *NeurIPS 2021* (2021) (page 11).

[HS17] Elias S Helou and Lucas EA Simões. "ϵ-subgradient algorithms for bilevel convex optimization". In: *Inverse Problems* 33.5 (2017), p. 055020 (page 4).

[JAMH23] Ruichen Jiang, Nazanin Abolfazli, Aryan Mokhtari, and Erfan Yazdandoost Hamedani. "A conditional gradient-based method for simple bilevel optimization with convex lower-level problem". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2023, pp. 10305–10323 (pages 2, 3, 5, 6, 9, 11).

[KY21] Harshal D Kaushik and Farzad Yousefian. "A method with convergence rates for optimization problems with variational inequality constraints". In: *SIAM Journal on Optimization* 31.3 (2021), pp. 2171–2198 (pages 2, 3, 11).

[LP94] Zhi-Quan Luo and Jong-Shi Pang. "Error bounds for analytic systems and their applications". In: *Mathematical Programming* 67.1-3 (1994), pp. 1–28 (page 9).

[Mal17] Yura Malitsky. "Chambolle-pock and tseng's methods: relationship and extension to the bilevel optimization". In: *arXiv preprint arXiv:1706.02602* (2017), p. 3 (page 5).

[MS23] Roey Merchav and Shoham Sabach. "Convex bi-level optimization problems with non-smooth outer objective function". In: *arXiv preprint arXiv:2307.08245* (2023) (pages 2, 3, 5, 11).

[NNG19] Ion Necoara, Yu Nesterov, and Francois Glineur. "Linear convergence of first order methods for non-strongly convex optimization". In: *Mathematical Programming* 175 (2019), pp. 69–107 (page 13).

[Pan97] Jong-Shi Pang. "Error bounds in mathematical programming". In: *Mathematical Programming* 79.1-3 (1997), pp. 299–332 (page 9).

[Pol87] Boris T Polyak. "Introduction to optimization". In: (1987) (page 28).

[PC17] Nelly Pustelnik and Laurent Condat. "Proximity operator of a sum of functions; application to depth map estimation". In: *IEEE Signal Processing Letters* 24.12 (2017), pp. 1827–1831 (page 24).

[RFKL19] Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. "Meta-learning with implicit gradients". In: *Advances in neural information processing systems* 32 (2019) (page 2).

[Rd17] Vincent Roulet and Alexandre d'Aspremont. "Sharpness, restart and acceleration". In: *Advances in Neural Information Processing Systems* 30 (2017) (page 9).

[Roz+21] Benedek Rozemberczki, Paul Scherer, Yixuan He, George Panagopoulos, Alexander Riedel, Maria Astefanoaei, Oliver Kiss, Ferenc Beres, Guzmán López, Nicolas Collignon, et al. "Pytorch geometric temporal: Spatiotemporal signal processing with neural machine learning models". In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2021, pp. 4564–4573 (pages 12, 28).

[SS17] Shoham Sabach and Shimrit Shtern. "A first order method for solving convex bilevel optimization problems". In: *SIAM Journal on Optimization* 27.2 (2017), pp. 640–660 (pages 5, 12, 13).

[SBY23] Sepideh Samadi, Daniel Burbano, and Farzad Yousefian. "Achieving optimal complexity guarantees for a class of bilevel convex optimization problems". In: *arXiv preprint arXiv:2310.12247* (2023) (pages 2–5, 8, 11, 24).

[SCHB19] Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. "Truncated back-propagation for bilevel optimization". In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 1723–1732 (page 2).

[SVZ21] Yekini Shehu, Phan Tu Vuong, and Alain Zemkoho. "An inertial extrapolation method for convex simple bilevel optimization". In: *Optimization Methods and Software* 36.1 (2021), pp. 1–19 (page 2).

[SHK23] Lingqing Shen, Nam Ho-Nguyen, and Fatma Kılınç-Karzan. "An online convex optimization-based framework for convex bilevel optimization". In: *Mathematical Programming* 198.2 (2023), pp. 1519–1582 (pages 2, 3, 11–13).

[Sol07a] Mikhail Solodov. "An explicit descent method for bilevel convex optimization". In: *Journal of Convex Analysis* 14.2 (2007), p. 227 (page 4).

[Sol07b] Mikhail V Solodov. "A bundle method for a class of bilevel nonsmooth convex minimization problems". In: *SIAM Journal on Optimization* 18.1 (2007), pp. 242–259 (page 4).

[TA77] A.N. Tikhonov and V.Y. Arsenin. *Solutions of Ill-Posed Problems*. New York: Wiley, 1977 (page 4).

[Tse08] Paul Tseng. "On Accelerated Proximal Gradient Methods for Convex-Concave Optimization". In: *submitted to SIAM J. Optim.* (2008) (page 6).

[WSJ24] Jiulin Wang, Xu Shi, and Rujun Jiang. "Near-Optimal Convex Simple Bilevel Optimization with a Bisection Method". In: *arXiv preprint arXiv:2402.05415* (2024) (pages 3, 4, 11, 24).

[Yu13] Yao-Liang Yu. "On decomposing the proximal map". In: *Advances in neural information processing systems* 26 (2013) (page 24).

Appendix

A Proof of the Main Results

A.1 Proof of Theorem 4.1

To prove Theorem 4.1, we start with the following general lemma that holds for any choice of the step sizes $\{a_k\}$.

Lemma A.1. *Let $\{\mathbf{x}_k\}$ be the sequence of iterates generated by Algorithm 1 with stepsize $a_k > 0$ for $k \geq 0$. Then we have*

$$\begin{aligned} A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right) \\ \leq \left(\frac{L_f a_k^2}{2A_{k+1}} - \frac{1}{2}\right)\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \end{aligned} \tag{19}$$

$$A_{k+1}(g(\mathbf{x}_{k+1}) - g(\mathbf{x}^*)) - A_k(g(\mathbf{x}_k) - g(\mathbf{x}^*)) \leq a_k(g_k - g(\mathbf{x}^*)) + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2. \tag{20}$$

Proof of Lemma A.1. Let $\mathbf{x}^*$ be any optimal solution of (1). We first consider the upper-level objective f. Since f is convex, we have

$$f(\mathbf{y}_k) - f(\mathbf{x}^*) \leq \langle \nabla f(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}^* \rangle, \quad f(\mathbf{y}_k) - f(\mathbf{x}_k) \leq \langle \nabla f(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle. \tag{21}$$

Now given the update rule $A_{k+1} = A_k + a_k$, we can write

$$A_{k+1}(f(\mathbf{y}_k) - f(\mathbf{x}^*)) - A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) = a_k(f(\mathbf{y}_k) - f(\mathbf{x}^*)) + A_k(f(\mathbf{y}_k) - f(\mathbf{x}_k)) \tag{22}$$

Combining (21) and (22), we have

$$\begin{aligned} A_{k+1}(f(\mathbf{y}_k) - f(\mathbf{x}^*)) - A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) &\leq a_k(\langle \nabla f(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}^* \rangle) + A_k(\langle \nabla f(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle) \\ &= \langle \nabla f(\mathbf{y}_k), a_k\mathbf{y}_k + A_k(\mathbf{y}_k - \mathbf{x}_k) - a_k\mathbf{x}^* \rangle \\ &= a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle, \end{aligned} \tag{23}$$

where the last equality follows from the definition of $\mathbf{y}_k$. Furthermore, since f is L_f-smooth, we have

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{y}_k) + \langle \nabla f(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_f}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2. \tag{24}$$

If we multiply both sides of (24) by A_{k+1} and combine the resulting inequality with (23), we obtain

$$\begin{aligned} &A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) - A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) \\ &\leq a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle + A_{k+1} \langle \nabla f(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_f A_{k+1}}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2 \\ &= a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle + a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{z}_k \rangle + \frac{L_f a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \\ &= a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle + \frac{L_f a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2, \end{aligned} \tag{25}$$

where we used the fact that $a_k(\mathbf{z}_{k+1}-\mathbf{z}_k) = A_{k+1}(\mathbf{x}_{k+1}-\mathbf{y}_k)$ in the first equality. Moreover, since $\mathbf{x}^* \in \mathcal{Z}_k$, we obtain from the update rule in (4) that

$$\begin{aligned} & \langle \mathbf{z}_{k+1} - \mathbf{z}_k + a_k \nabla f(\mathbf{y}_k), \mathbf{x}^* - \mathbf{z}_{k+1} \rangle \geq 0 \\ \Leftrightarrow \quad & a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle \leq \langle \mathbf{z}_{k+1} - \mathbf{z}_k, \mathbf{x}^* - \mathbf{z}_{k+1} \rangle \\ \Leftrightarrow \quad & a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle \leq \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2 - \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2. \end{aligned} \tag{26}$$

Combining (25) and (26) leads to

$$\begin{aligned} A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2 \right) \\ \leq \frac{1}{2}\left(\frac{L_f a_k^2}{A_{k+1}} - 1 \right) \|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2, \end{aligned}$$

which proves the claim in (19).

Next, we proceed to prove the claim in (20). To do so, we first leverage the convexity of g which leads to

$$g(\mathbf{y}_k) - g(\mathbf{x}_k) \leq \langle \nabla g(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle . \tag{27}$$

Also, since g is L_g-smooth, we have

$$g(\mathbf{x}_{k+1}) \leq g(\mathbf{y}_k) + \langle \nabla g(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_g}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2. \tag{28}$$

By multiplying both sides of (27) and (28) by A_k and A_{k+1}, respectively, and adding the resulted inequalities we obtain

$$\begin{aligned} & A_{k+1}(g(\mathbf{x}_{k+1}) - g(\mathbf{y}_k)) + A_k(g(\mathbf{y}_k) - g(\mathbf{x}_k)) \\ & \leq A_{k+1} \langle \nabla g(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + A_k \langle \nabla g(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle + \frac{L_g A_{k+1}}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2 \\ & = a_k \langle \nabla g(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{z}_k \rangle + A_k \langle \nabla g(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \\ & = a_k \langle \nabla g(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{y}_k \rangle + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2, \end{aligned}$$

where the first equality holds since $a_k(\mathbf{z}_{k+1} - \mathbf{z}_k) = A_{k+1}(\mathbf{x}_{k+1} - \mathbf{y}_k)$, and the second equality holds since $a_k(\mathbf{z}_k - \mathbf{y}_k) = A_k(\mathbf{y}_k - \mathbf{x}_k)$. Lastly, by the definition of the constructed cutting plane, we know that $g(\mathbf{y}_k) + \langle \nabla g(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle \leq g_k$ for any $\mathbf{z}$. Hence, $\langle \nabla g(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{y}_k \rangle$ is upper bounded by $g_k - g(\mathbf{y}_k)$ which itself is upper bounded by $g_k - g(\mathbf{x}^*)$. Applying this substitution into to the above expression would lead to the claim in (20). □

Now we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. To begin with, note that by our choice of a_k, we have

$$a_k = \frac{k+1}{4L_f} \quad \text{and} \quad A_{k+1} = \frac{(k+1)(k+2)}{8L_f}. \tag{29}$$

Thus, it can be verified that $L_f a_k^2 \leq \frac{1}{2} A_{k+1}$. Then it follows from Lemma A.1 that

$$A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right) \leq -\frac{1}{4}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \tag{30}$$

$$A_{k+1}(g(\mathbf{x}_{k+1}) - g(\mathbf{x}^*)) - A_k(g(\mathbf{x}_k) - g(\mathbf{x}^*)) \leq a_k(g_k - g(\mathbf{x}^*)) + \frac{L_g}{4L_f}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2. \tag{31}$$

We first prove the convergence guarantee for the upper-level objective. By using induction on (30), we obtain that for any $k \geq 0$

$$A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2 \leq A_0(f(\mathbf{x}_0) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_0 - \mathbf{x}^*\|^2 = \frac{1}{2}\|\mathbf{z}_0 - \mathbf{x}^*\|^2, \tag{32}$$

which implies

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{2A_k} = \frac{4L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{k(k+1)}.$$

We proceed to establish an upper bound on $g(\mathbf{x}_k) - g(\mathbf{x}^*)$. By summing the inequality in (31) from 0 to $k-1$ we obtain

$$\begin{aligned} A_k(g(\mathbf{x}_k) - g(\mathbf{x}^*)) &\leq \sum_{i=0}^{k-1} a_i(g_i - g^*) + \frac{L_g}{4L_f}\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2 \\ &\leq \sum_{i=0}^{k-1} \frac{i+1}{4L_f}\frac{2L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{(i+1)^2} + \frac{L_g}{4L_f}\sum_{i=0}^{k-1} D^2 \\ &= \frac{L_g}{2L_f}\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1) + \frac{L_g}{4L_f}D^2 k. \end{aligned} \tag{33}$$

Note that the second inequality holds due to the condition in (8). Thus, we obtain

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)} + \frac{2L_g D^2}{k+1}.$$

The above upper bound on $g(\mathbf{x}_k) - g(\mathbf{x}^*)$ without any additional condition, but next we show that if $f(\mathbf{x}_k) \geq f(\mathbf{x}^*)$ the above upper bound can be further improved as we can upper bound $\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2$ by a constant independent of k instead of kD^2. To prove this claim, by summing the inequality in (19) from 0 to $k-1$ we obtain

$$\frac{1}{4}\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2 \leq \frac{1}{2}\|\mathbf{z}_0 - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right). \tag{34}$$

Hence, if $f(\mathbf{x}_k) \geq f(\mathbf{x}^*)$, then it holds

$$\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2 \leq 2\|\mathbf{z}_0 - \mathbf{x}^*\|^2. \tag{35}$$

Thus if replace $\sum_{i=0}^{k-1}\|\mathbf{z}_{i+1} - \mathbf{z}_i\|^2$ in (33) by $2\|\mathbf{z}_0 - \mathbf{x}^*\|^2$, we would obtain the following improve bound:

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)} + \frac{4L_g\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{k(k+1)}.$$

□

Recall Remark 4.1. The main difficulty in obtaining an accelerated rate of $\mathcal{O}(1/K^2)$ for g is that it is unclear how to control $\sum_{k=0}^{K-1}\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2$. This, in turn, is because we don't know how to prove a **lower bound** on f. Instead, we used the compactness of $\mathcal{Z}$ to achieve the $\mathcal{O}(1/K)$ for g.

A.2 Proof of Lemma 4.3

Proof of Lemma 4.3. Note that by multiplying both sides of (19) by $\lambda > 0$ we have

$$A_{k+1}(\lambda(f(\mathbf{x}_{k+1})-f(\mathbf{x}^*)))+\frac{\lambda}{2}\|\mathbf{z}_{k+1}-\mathbf{x}^*\|^2-\left(A_k(\lambda(f(\mathbf{x}_k)-f(\mathbf{x}^*)))+\frac{\lambda}{2}\|\mathbf{z}_k-\mathbf{x}^*\|^2\right) \\ \leq \frac{\lambda}{2}\left(\frac{L_f a_k^2}{A_{k+1}}-1\right)\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \tag{36}$$

Further note that in this case we have $a_k=\frac{\gamma(k+1)}{4L_f}$ and $A_{k+1}=\gamma\frac{(k+1)(k+2)}{8L_f}$. Hence, a_k^2/A_{k+1} is bounded above by $\frac{\gamma}{2L_f}$. Therefore, we can replace a_k^2/A_{k+1} in the above expression by $\frac{\gamma}{2L_f}$ to obtain

$$A_{k+1}(\lambda(f(\mathbf{x}_{k+1})-f(\mathbf{x}^*)))+\frac{\lambda}{2}\|\mathbf{z}_{k+1}-\mathbf{x}^*\|^2-\left(A_k(\lambda(f(\mathbf{x}_k)-f(\mathbf{x}^*)))+\frac{\lambda}{2}\|\mathbf{z}_k-\mathbf{x}^*\|^2\right) \\ \leq \frac{\lambda}{2}\left(\frac{\gamma}{2}-1\right)\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \tag{37}$$

Similarly, we can replace a_k^2/A_{k+1} in (20) by $\frac{\gamma}{2L_f}$ to obtain

$$A_{k+1}(g(\mathbf{x}_{k+1})-g(\mathbf{x}^*))-A_k(g(\mathbf{x}_k)-g(\mathbf{x}^*))\leq a_k(g_k-g(\mathbf{x}^*))+\frac{\gamma L_g}{4L_f}\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \tag{38}$$

Note if we add above up the above inequalities in (36) and (37) we obtain

$$\begin{aligned} &A_{k+1}(\lambda(f(\mathbf{x}_{k+1})-f(\mathbf{x}^*))+g(\mathbf{x}_{k+1})-g(\mathbf{x}^*))+\frac{\lambda}{2}\|\mathbf{z}_{k+1}-\mathbf{x}^*\|^2 \\ &\quad-\left(A_k(\lambda(f(\mathbf{x}_k)-f(\mathbf{x}^*))+g(\mathbf{x}_k)-g(\mathbf{x}^*))+\frac{\lambda}{2}\|\mathbf{z}_k-\mathbf{x}^*\|^2\right) \\ &\leq(\lambda(\frac{\gamma}{4}-\frac{1}{2})+\frac{\gamma}{4}\frac{L_g}{L_f})\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2+a_k(g_k-g(\mathbf{x}^*))\leq a_k(g_k-g(\mathbf{x}^*)) \end{aligned} \tag{39}$$

Note that the last inequality holds since the first term is negative due to the choice of λ. By summing the inequalities from 0 to $k-1$ we obtain

$$A_k(\lambda(f(\mathbf{x}_k)-f(\mathbf{x}^*))+g(\mathbf{x}_k)-g(\mathbf{x}^*))+\frac{1}{2}\lambda\|\mathbf{z}_k-\mathbf{x}^*\|^2\leq\sum_{i=0}^{k-1}a_i(g_i-g(\mathbf{x}^*))+\frac{1}{2}\lambda\|\mathbf{z}_0-\mathbf{x}^*\|^2 \tag{40}$$

Now using the condition on g_i in (8) and the definition of a_i we have

$$\sum_{i=0}^{k-1}a_i(g_i-g(\mathbf{x}^*))\leq\sum_{i=0}^{k-1}\frac{\gamma(i+1)}{4L_f}\frac{2L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2}{(i+1)^2}\leq\frac{\gamma L_g}{2L_f}\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln k+1) \tag{41}$$

By applying this upper bound into (40) we obtain

$$A_k(\lambda(f(\mathbf{x}_k)-f(\mathbf{x}^*))+g(\mathbf{x}_k)-g(\mathbf{x}^*))+\frac{1}{2}\lambda\|\mathbf{z}_k-\mathbf{x}^*\|^2\leq\frac{\gamma L_g}{2L_f}\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln k+1)+\frac{1}{2}\lambda\|\mathbf{z}_0-\mathbf{x}^*\|^2 \tag{42}$$

If we drop the $\frac{1}{2}\lambda\|\mathbf{z}_k - \mathbf{x}^*\|^2$ in the left-hand side and divide both sides of the resulted inequality by A_K which is equal to $A_k = \gamma\frac{k(k+1)}{8L_f}$ we obtain

$$\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln k + 1)}{k(k+1)} + \frac{4\lambda L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)}, \tag{43}$$

and the proof is complete. □

A.3 Proof of Theorem 4.4

Proof of Theorem 4.4. Recall the result of Lemma 4.3 that if $a_k = \gamma(k+1)/(4L_f)$, where $0 < \gamma \leq 1$ and $\lambda \geq \frac{L_g}{(2/\gamma-1)L_f}$ then after T iterations we have

$$\lambda(f(\mathbf{x}_T) - f(\mathbf{x}^*)) + g(\mathbf{x}_T) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{T(T+1)} + \frac{4\lambda L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{\gamma T(T+1)}. \tag{44}$$

Now if we replace γ by $1/(\frac{2L_g}{L_f}T^{\frac{2r-2}{2r-1}} + 2)$ as suggested in the statement of the theorem, we would obtain

$$\lambda(f(\mathbf{x}_T) - f(\mathbf{x}^*)) + g(\mathbf{x}_T) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{T(T+1)} + \frac{8(\frac{L_g}{L_f}T^{\frac{2r-2}{2r-1}} + 1)\lambda L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T(T+1)}. \tag{45}$$

Now we proceed to prove the first claim which is an upper bound on $f(\mathbf{x}_T) - f(\mathbf{x}^*)$. Note that given the fact that $g(\mathbf{x}_T) - g(\mathbf{x}^*) > 0$ and $\lambda > 0$ we can show that

$$f(\mathbf{x}_T) - f(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{\lambda T(T+1)} + \frac{8((\frac{L_g}{L_f}T^{\frac{2r-2}{2r-1}} + 1))L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T(T+1)}. \tag{46}$$

If we select λ which is a free parameter as $\lambda = T^{-\frac{2r-2}{2r-1}} \geq \frac{L_g}{(2/\gamma-1)L_f}$ then we obtain

$$f(\mathbf{x}_T) - f(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{T^{\frac{1}{2r-1}}(T+1)} + \frac{8L_g\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T^{\frac{1}{2r-1}}(T+1)} + \frac{8L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T(T+1)}. \tag{47}$$

Given the fact that $\mathbf{x}_0 = \mathbf{z}_0$ we can simplify the upper bound to

$$f(\mathbf{x}_T) - f(\mathbf{x}^*) \leq \frac{12L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{T^{\frac{2r}{2r-1}}} + \frac{8L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T^2}. \tag{48}$$

Next, we proceed to establish an upper bound on $g(\mathbf{x}_T) - g(\mathbf{x}^*)$. We will use the following inequality that holds due to the HEB condition and formally stated in Proposition 4.2:

$$f(\mathbf{x}_T) - f^* \geq -M\left(\frac{r(g(\mathbf{x}_T) - g(\mathbf{x}^*)))}{\alpha}\right)^{\frac{1}{r}} \tag{49}$$

Now if we replace this lower bound into (45) we would obtain

$$\begin{aligned} &-\lambda M(\frac{r}{\alpha})^{\frac{1}{r}}(g(\mathbf{x}_T) - g(\mathbf{x}^*))^{\frac{1}{r}} + g(\mathbf{x}_T) - g(\mathbf{x}^*) \\ &\leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2(\ln T + 1)}{T(T+1)} + \frac{8((\frac{L_g}{L_f}T^{\frac{2r-2}{2r-1}} + 1))\lambda L_f\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{T(T+1)}. \end{aligned} \tag{50}$$

Next we consider two different cases: In the first case we assume $\lambda M(\frac{r}{\alpha})^{\frac{1}{r}}(g(\mathbf{x}_T)-g(\mathbf{x}^*))^{\frac{1}{r}} \leq \frac{1}{2}(g(\mathbf{x}_T)-g(\mathbf{x}^*))$ holds and in the second case we assume the opposite of this inequality holds.

If we are in the first case and $\lambda M(\frac{r}{\alpha})^{\frac{1}{r}}(g(\mathbf{x}_T)-g(\mathbf{x}^*))^{\frac{1}{r}} \leq \frac{1}{2}(g(\mathbf{x}_T)-g(\mathbf{x}^*))$, then the inequality in (50) leads to

$$\frac{1}{2}(g(\mathbf{x}_T)-g(\mathbf{x}^*)) \leq \frac{4L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1)}{T(T+1)} + \frac{8((\frac{L_g}{L_f}T^{\frac{2r-2}{2r-1}}+1))\lambda L_f\|\mathbf{z}_0-\mathbf{x}^*\|^2}{T(T+1)}. \tag{51}$$

Since $\lambda = T^{-\frac{2r-2}{2r-1}}$, it further leads to the following upper bound

$$g(\mathbf{x}_T)-g(\mathbf{x}^*) \leq \frac{8L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1)}{T(T+1)} + \frac{16L_g\|\mathbf{z}_0-\mathbf{x}^*\|^2}{T(T+1)} + \frac{16L_f\|\mathbf{z}_0-\mathbf{x}^*\|^2}{T^{\frac{1}{2r-1}}(T+1)}. \tag{52}$$

Now given the fact that $\mathbf{x}_0=\mathbf{z}_0$, we obtain

$$g(\mathbf{x}_T)-g(\mathbf{x}^*) \leq \frac{24L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1)}{T^2} + \frac{16L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2}{T^{\frac{2r}{2r-1}}}. \tag{53}$$

If we are in the second case and $\lambda M(\frac{r}{\alpha})^{\frac{1}{r}}(g(\mathbf{x}_T)-g(\mathbf{x}^*))^{\frac{1}{r}} > \frac{1}{2}(g(\mathbf{x}_T)-g(\mathbf{x}^*))$ then this inequality is equivalent to

$$(g(\mathbf{x}_T)-g(\mathbf{x}^*))^{1-1/r} \leq 2\lambda M(\frac{r}{\alpha})^{\frac{1}{r}}, \tag{54}$$

leading to

$$g(\mathbf{x}_T)-g(\mathbf{x}^*) \leq \left(2T^{-\frac{2r-2}{2r-1}}M(\frac{r}{\alpha})^{\frac{1}{r}}\right)^{\frac{r}{r-1}} = \frac{(2M)^{\frac{r}{r-1}}(\frac{r}{\alpha})^{\frac{1}{r-1}}}{T^{\frac{2r}{2r-1}}} \tag{55}$$

By combining the bounds in (53) and (55) we realize that

$$g(\mathbf{x}_T)-g(\mathbf{x}^*) \leq \max\{\frac{24L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1)}{T^2} + \frac{16L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2}{T^{\frac{2r}{2r-1}}}, \frac{(2M)^{\frac{r}{r-1}}(\frac{r}{\alpha})^{\frac{1}{r-1}}}{T^{\frac{2r}{2r-1}}}\} \tag{56}$$

Finally by using the above bound in (56) and the result of Proposition 4.2 we can prove the the second claim and establish a lower bound on $f(\mathbf{x}_T)-f^*$ which is

$$f(\mathbf{x}_T)-f^* \geq -M\left(\frac{r}{\alpha}\right)^{\frac{1}{r}}\left(\max\{\frac{24L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1)}{T^2} + \frac{16L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2}{T^{\frac{2r}{2r-1}}}, \frac{(2M)^{\frac{r}{r-1}}(\frac{r}{\alpha})^{\frac{1}{r-1}}}{T^{\frac{2r}{2r-1}}}\}\right)^{1/r} \tag{57}$$

leading to

$$f(\mathbf{x}_T)-f^* \geq \\ -M\left(\frac{r}{\alpha}\right)^{\frac{1}{r}}\left(\max\{\frac{(24L_g\|\mathbf{x}_0-\mathbf{x}^*\|^2(\ln T+1))^{1/r}}{T^{2/r}} + \frac{(16L_f\|\mathbf{x}_0-\mathbf{x}^*\|^2)^{1/r}}{T^{\frac{2}{2r-1}}}, \frac{((2M)^{\frac{r}{r-1}}(\frac{r}{\alpha})^{\frac{1}{r-1}})^{1/r}}{T^{\frac{2}{2r-1}}}\}\right) \tag{58}$$

□

A.4 Proof of Theorem 4.5

Proof of Theorem 4.5. To upper bound $f(\mathbf{x}_k)-f(\mathbf{x}^*)$, we follow a similar analysis as in Theorem 4.1. Specifically, first note that by our choice of a_k, we have $a_k = \gamma\frac{k+1}{4L_f}$ and $A_{k+1} = \gamma\frac{(k+1)(k+2)}{8L_f}$, where $\gamma \in (0,1)$. Hence, we can obtain that $L_f a_k^2 \leq \frac{\gamma}{2}A_{k+1}$. By using Lemma A.1 and the fact that $\gamma \in (0,1)$, we have

$$A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right) \leq \left(\frac{\gamma}{4} - \frac{1}{2}\right)\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \leq 0.$$

By using induction, we obtain that for any $k \geq 0$

$$A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2 \leq A_0(f(\mathbf{x}_0) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_0 - \mathbf{x}^*\|^2 = \frac{1}{2}\|\mathbf{z}_0 - \mathbf{x}^*\|^2, \tag{59}$$

Since $A_k = \gamma\frac{k(k+1)}{8L_f}$ and $\mathbf{z}_0 = \mathbf{x}_0$, this further implies that

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{2A_k} = \frac{4L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)}.$$

Next, we will prove the upper bound on $g(\mathbf{x}_k) - g(\mathbf{x}^*)$. By Lemma 4.3, we have for any $k \geq 0$

$$\lambda(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{k(k+1)}(\ln k + 1) + \frac{4\lambda L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)} \tag{60}$$

Moreover, since g satisfies the weak sharpness condition, we can use Proposition 4.2 with $r = 1$ to write

$$f(\mathbf{x}_k) - f^* \geq -\frac{M}{\alpha}(g(\mathbf{x}_k) - g^*). \tag{61}$$

Combining (60) and (61) leads to

$$-\lambda\frac{M}{\alpha}(g(\mathbf{x}_k) - g(\mathbf{x}^*)) + g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{4L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{k(k+1)}(\ln k + 1) + \frac{4\lambda L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)} \tag{62}$$

Note that we can choose λ to be any number satisfying $\lambda \geq \frac{L_g}{(2/\gamma-1)L_f}$ (cf. Lemma 4.3). Specifically, since $\gamma \leq \frac{2\alpha L_f}{2ML_g+\alpha L_f}$, we can set $\lambda = \alpha/(2M)$ and accordingly (62) can be simplified to

$$g(\mathbf{x}_k) - g(\mathbf{x}^*) \leq \frac{8L_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{k(k+1)}(\ln k + 1) + \frac{4\alpha L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma M k(k+1)}.$$

Finally, we use (61) again together with the above upper bound on $g(\mathbf{x}_k) - g(\mathbf{x}^*)$ to obtain

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \geq -\frac{M}{\alpha}(g(\mathbf{x}_k) - g(\mathbf{x}^*)) \geq -\left(\frac{8ML_g\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\alpha k(k+1)}(\ln k + 1) + \frac{4L_f\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{\gamma k(k+1)}\right).$$

□

B Extension to the Non-smooth/Composite Setting

In this section, we would like to mention the possible extension to the non-smooth/composite setting. In the general non-smooth settings, we believe it is not possible to extend our results and achieve the purpose of the acceleration. This is because, even in the single-level setting, the best achievable rate in the general non-smooth setting is $\mathcal{O}(1/\sqrt{K})$ achieved by sub-gradient method. That said, it should be possible to extend our accelerated bilevel framework to a special non-smooth setting where the upper- and lower-level objective functions have a composite structure, i.e., they can be written as the sum of a convex smooth function and a convex non-smooth function that is easy to compute its proximal operator.

Then we consider the composite counterpart of Problem (1):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; f(\mathbf{x}) := f_1(\mathbf{x}) + f_2(\mathbf{x}) \qquad \text{s.t.} \quad \mathbf{x} \in \operatorname*{argmin}_{\mathbf{z}\in\mathbb{R}^n} \; g(\mathbf{z}) := g_1(\mathbf{z}) + g_2(\mathbf{z}), \tag{63}$$

where $f_1, g_1 : \mathbb{R}^n \to \mathbb{R}$ are smooth convex functions and $f_2, g_2 : \mathbb{R}^n \to \mathbb{R}$ are nonsmooth convex functions, respectively.

To analyze and implement the proximal gradient-based methods, we need the following definition concerning the property of the proximal mapping.

Definition B.1. *Given a function $h : \mathbb{R}^n \to (-\infty, +\infty]$, the proximal map of h is defined for all $\mathbf{x} \in \mathbb{R}^n$ and $\eta > 0$ as*

$$Prox_{\eta h}(\mathbf{x}) \triangleq \operatorname*{argmin}_{\mathbf{u}\in\mathbb{R}^n} \{\frac{1}{2\eta}\|\mathbf{u}-\mathbf{x}\|^2 + h(\mathbf{u})\} \tag{64}$$

To handle the upper-level nonsmooth part f_2, we need to change the projection step in Step 6 of Algorithm 1 to a proximal update (outlined in the Step 6 of Algorithm 2), which is similar to the accelerated proximal gradient method for single-level problems in [BT09]. On the other hand, to deal with the lower-level nonsmooth part g_2, it is necessary to modify the approximated set of the lower-level solution set $\mathcal{X}_k$. Specifically, since g_2 is nonsmooth, ∇g may not be obtained over the whole domain. Hence, we utilize g_1 and g_2 separately in the construction of $\mathcal{X}_k$ by summing a linear approximation of g_1 and g_2 up as a lower bound of g_k. Note that the constructed set $\mathcal{X}_k$ is no longer a hyperplane in this setting due to the possibly non-linear nature of g_2. We refer to our method as the Proximal Accelerated Gradient Method for Bilevel Optimization (P-AGM-BiO) and its steps are outlined in Algorithm 2.

Different proximal-friendly assumptions are commonly used in the literature of composite single-level/bilevel optimization [BT09; SBY23; WSJ24; CSJW24]. The following proximal-friendly assumption is necessary for our method in the composite setting.

Assumption B.1. *The function $f_2 + \delta_{\mathcal{X}_k}$ in the Step 6 of Algorithm 2 is proximal-friendly, i.e. the proximal mapping in Definition B.1 is easy to compute, for all k, where $\delta_{\mathcal{X}_k}(\cdot)$ is the indicator function.*

This assumption implies that f_2 is proximal-friendly and projecting onto the constructed set $\mathcal{X}_k$ is easy. Moreover, The function $f_2+\delta_{\mathcal{X}_k}$ is the sum of two convex functions, and the study of proximal mapping for sums of functions can be found in the literature [Yu13; PC17; BBW18; ABC19].

Note that the properties of smoothness of f and g have only been used in the proof of Lemma A.1 in Section 4. None of the other results will break if (19) and (20) in Lemma A.1 still hold in the

Algorithm 2 Proximal Accelerated Gradient Method for Bilevel Optimization (P-AGM-BiO)

Input: A sequence $\{g_k\}_{k=0}^K$, a scalar $\gamma \in (0, 1]$
Initialization: $A_0 = 0$, $\mathbf{x}_0 = \mathbf{z}_0 \in \mathbb{R}^n$
for $k = 0, \ldots, K$ **do**
  Set $a_k = \gamma \dfrac{k+1}{4L_f}$
  Compute $\mathbf{y}_k = \dfrac{A_k}{A_k + a_k}\mathbf{x}_k + \dfrac{a_k}{A_k + a_k}\mathbf{z}_k$
  Compute $\mathbf{z}_{k+1} = Prox_{a_k(f_2+\delta_{\mathcal{X}_k})}(\mathbf{z}_k - a_k \nabla f_1(\mathbf{y}_k))$, where

$$\mathcal{X}_k \triangleq \{\mathbf{z} \in \mathbb{R}^n : g_1(\mathbf{y}_k) + \langle \nabla g_1(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle + g_2(\mathbf{z}) \leq g_k\}$$

  Compute $\mathbf{x}_{k+1} = \dfrac{A_k}{A_k + a_k}\mathbf{x}_k + \dfrac{a_k}{A_k + a_k}\mathbf{z}_{k+1}$
  Update $A_{k+1} = A_k + a_k$
end for
Return: $\mathbf{x}_K$

composite setting. Now, we present and prove the counterpart of Lemma A.1 in the composite setting.

Lemma B.1. *Suppose f_1, f_2, g_1, g_2 are convex and f_1, g_1 are L_f-smooth and L_g-smooth, respectively. Let $\{\mathbf{x}_k\}$ be the sequence of iterates generated by Algorithm 2 with stepsize $a_k > 0$ for $k \geq 0$. Moreover, suppose Assumption B.1 holds. Then we have*

$$\begin{aligned} A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \Big(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\Big) \\ \leq \left(\frac{L_f a_k^2}{2A_{k+1}} - \frac{1}{2}\right)\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2, \end{aligned} \tag{65}$$

$$A_{k+1}(g(\mathbf{x}_{k+1}) - g(\mathbf{x}^*)) - A_k(g(\mathbf{x}_k) - g(\mathbf{x}^*)) \leq a_k(g_k - g(\mathbf{x}^*)) + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2. \tag{66}$$

Proof of Lemma B.1. Let $\mathbf{x}^*$ be any optimal solution of (63).

We first consider the upper-level objective f. Since f_1 is convex, we have

$$f_1(\mathbf{y}_k) - f_1(\mathbf{x}^*) \leq \langle \nabla f_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}^* \rangle, \quad f_1(\mathbf{y}_k) - f_1(\mathbf{x}_k) \leq \langle \nabla f_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle. \tag{67}$$

Now given the update rule $A_{k+1} = A_k + a_k$, we can write

$$A_{k+1}(f_1(\mathbf{y}_k) - f_1(\mathbf{x}^*)) - A_k(f_1(\mathbf{x}_k) - f_1(\mathbf{x}^*)) = a_k(f_1(\mathbf{y}_k) - f_1(\mathbf{x}^*)) + A_k(f_1(\mathbf{y}_k) - f_1(\mathbf{x}_k)) \tag{68}$$

Combining (67) and (68), we have

$$\begin{aligned} A_{k+1}(f_1(\mathbf{y}_k) - f_1(\mathbf{x}^*)) - A_k(f_1(\mathbf{x}_k) - f_1(\mathbf{x}^*)) &\leq a_k(\langle \nabla f_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}^* \rangle) + A_k(\langle \nabla f_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle) \\ &= \langle \nabla f_1(\mathbf{y}_k), a_k\mathbf{y}_k + A_k(\mathbf{y}_k - \mathbf{x}_k) - a_k\mathbf{x}^* \rangle \\ &= a_k \langle \nabla f_1(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle, \end{aligned} \tag{69}$$

where the last equality follows from the definition of $\mathbf{y}_k$. Furthermore, since f_1 is L_f-smooth, we have

$$f_1(\mathbf{x}_{k+1}) \leq f_1(\mathbf{y}_k) + \langle \nabla f_1(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_f}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2. \tag{70}$$

If we multiply both sides of (70) by A_{k+1} and combine the resulting inequality with (69), we obtain

$$\begin{aligned}
&A_{k+1}(f_1(\mathbf{x}_{k+1}) - f_1(\mathbf{x}^*)) - A_k(f_1(\mathbf{x}_k) - f_1(\mathbf{x}^*)) \\
&\leq a_k \langle \nabla f_1(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle + A_{k+1} \langle \nabla f_1(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_f A_{k+1}}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2 \\
&= a_k \langle \nabla f_1(\mathbf{y}_k), \mathbf{z}_k - \mathbf{x}^* \rangle + a_k \langle \nabla f_1(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{z}_k \rangle + \frac{L_f a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \\
&= a_k \langle \nabla f_1(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle + \frac{L_f a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2,
\end{aligned} \tag{71}$$

where we used the fact that $a_k(\mathbf{z}_{k+1} - \mathbf{z}_k) = A_{k+1}(\mathbf{x}_{k+1} - \mathbf{y}_k)$ in the first equality. Moreover, from the step 6 in Algorithm 2, we have $\mathbf{z}_k - a_k \nabla f_1(\mathbf{y}_k) - \mathbf{z}_{k+1} \in a_k \partial(f_2(\mathbf{z}_{k+1}) + \delta_{\mathcal{X}_k}(\mathbf{z}_{k+1}))$. Using this, from the definition of subgradients for $f_2 + \delta_{\mathcal{X}_k}$, we have

$$\begin{aligned}
&\langle \mathbf{x}^* - \mathbf{z}_{k+1}, \mathbf{z}_k - a_k \nabla f_1(\mathbf{y}_k) - \mathbf{z}_{k+1} \rangle \leq a_k(f_2(\mathbf{x}^*) + \delta_{\mathcal{X}_k}(\mathbf{x}^*) - f_2(\mathbf{z}_{k+1}) - \delta_{\mathcal{X}_k}(\mathbf{z}_{k+1})) \\
\Leftrightarrow \quad &\langle \mathbf{z}_{k+1} - \mathbf{z}_k + a_k \nabla f(\mathbf{y}_k), \mathbf{x}^* - \mathbf{z}_{k+1} \rangle \geq a_k f_2(\mathbf{z}_{k+1}) - a_k f_2(\mathbf{x}^*) \\
\Leftrightarrow \quad &a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle \leq \langle \mathbf{z}_{k+1} - \mathbf{z}_k, \mathbf{x}^* - \mathbf{z}_{k+1} \rangle - a_k f_2(\mathbf{z}_{k+1}) + a_k f_2(\mathbf{x}^*) \\
\Leftrightarrow \quad &a_k \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x}^* \rangle \leq \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2 - \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \\
&\qquad\qquad - a_k f_2(\mathbf{z}_{k+1}) + a_k f_2(\mathbf{x}^*).
\end{aligned} \tag{72}$$

The first step holds since $\mathbf{x}^*, \mathbf{z}_{k+1} \in \mathcal{X}_k$, i.e. $\delta_{\mathcal{X}_k}(\mathbf{x}^*) = \delta_{\mathcal{X}_k}(\mathbf{z}_{k+1}) = 0$. Combining (71) and (72) leads to

$$\begin{aligned}
&A_{k+1}(f_1(\mathbf{x}_{k+1}) - f_1(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f_1(\mathbf{x}_k) - f_1(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right) \\
&\leq \frac{1}{2}\left(\frac{L_f a_k^2}{A_{k+1}} - 1\right)\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 - a_k f_2(\mathbf{z}_{k+1}) + a_k f_2(\mathbf{x}^*),
\end{aligned} \tag{73}$$

Then we add $(A_{k+1}f_2(\mathbf{x}_{k+1}) - a_k f_2(\mathbf{x}^*) - A_k f_2(\mathbf{x}_k))$ on both sides to obtain,

$$\begin{aligned}
&A_{k+1}(f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_{k+1} - \mathbf{x}^*\|^2 - \left(A_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \frac{1}{2}\|\mathbf{z}_k - \mathbf{x}^*\|^2\right) \\
&\leq \frac{1}{2}\left(\frac{L_f a_k^2}{A_{k+1}} - 1\right)\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 - a_k f_2(\mathbf{z}_{k+1}) + A_{k+1}f_2(\mathbf{x}_{k+1}) - A_k f_2(\mathbf{x}_k),
\end{aligned} \tag{74}$$

Finally, by the convexity of f_2, $A_{k+1} = A_k + a_k$, and $\mathbf{x}_{k+1} = \frac{A_k}{A_k + a_k}\mathbf{x}_k + \frac{a_k}{A_k + a_k}\mathbf{z}_{k+1}$, i.e. $-a_k f_2(\mathbf{z}_{k+1}) + A_{k+1}f_2(\mathbf{x}_{k+1}) - A_k f_2(\mathbf{x}_k) \leq 0$, the first inequality of this Lemma can be obtained.

Next, we proceed to prove the claim for the lower-level objective g. To do so, we first leverage the convexity of the smooth part g_1 which leads to

$$g_1(\mathbf{y}_k) - g_1(\mathbf{x}_k) \leq \langle \nabla g_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \rangle. \tag{75}$$

Also, since g_1 is L_g-smooth, we have

$$g_1(\mathbf{x}_{k+1}) \leq g_1(\mathbf{y}_k) + \langle \nabla g_1(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \rangle + \frac{L_g}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2. \tag{76}$$

By multiplying both sides of (75) and (76) by A_k and A_{k+1}, respectively, and adding the resulted inequalities we obtain

$$\begin{aligned}
&A_{k+1}(g_1(\mathbf{x}_{k+1}) - g_1(\mathbf{y}_k)) + A_k(g_1(\mathbf{y}_k) - g_1(\mathbf{x}_k)) \\
&\leq A_{k+1}\left\langle \nabla g_1(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{y}_k \right\rangle + A_k\left\langle \nabla g_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \right\rangle + \frac{L_g A_{k+1}}{2}\|\mathbf{x}_{k+1} - \mathbf{y}_k\|^2 \\
&= a_k\left\langle \nabla g_1(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{z}_k \right\rangle + A_k\left\langle \nabla g_1(\mathbf{y}_k), \mathbf{y}_k - \mathbf{x}_k \right\rangle + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \\
&= a_k\left\langle \nabla g_1(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{y}_k \right\rangle + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2,
\end{aligned}$$

where the first equality holds since $a_k(\mathbf{z}_{k+1} - \mathbf{z}_k) = A_{k+1}(\mathbf{x}_{k+1} - \mathbf{y}_k)$, and the second equality holds since $a_k(\mathbf{z}_k - \mathbf{y}_k) = A_k(\mathbf{y}_k - \mathbf{x}_k)$. Lastly, by the definition of the constructed approximated set $\mathcal{X}_k$, we know that $g_1(\mathbf{y}_k) + \langle \nabla g_1(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle + g_2(\mathbf{z}) \leq g_k$ for any $\mathbf{z} \in \mathcal{X}_k$. Hence, $\langle \nabla g_1(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{y}_k \rangle$ is upper bounded by $g_k - g_1(\mathbf{y}_k) - g_2(\mathbf{z}_{k+1})$. Applying this substitution into to the above expression to obtain,

$$\begin{aligned}
&A_{k+1}(g_1(\mathbf{x}_{k+1}) - g_1(\mathbf{y}_k)) + A_k(g_1(\mathbf{y}_k) - g_1(\mathbf{x}_k)) \\
&\leq a_k g_k - a_k g_1(\mathbf{y}_k) - a_k g_2(\mathbf{z}_{k+1}) + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2
\end{aligned} \tag{77}$$

By adding $a_k g_1(\mathbf{y}_k) - a_k g_1(\mathbf{x}^*)$ on both sides, we have,

$$\begin{aligned}
&A_{k+1}(g_1(\mathbf{x}_{k+1}) - g_1(\mathbf{x}^*)) + A_k(g_1(\mathbf{x}^*) - g_1(\mathbf{x}_k)) \\
&\leq a_k g_k - a_k g_1(\mathbf{x}^*) - a_k g_2(\mathbf{z}_{k+1}) + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2
\end{aligned} \tag{78}$$

Lastly, we add $(A_{k+1}g_2(\mathbf{x}_{k+1}) - a_k g_2(\mathbf{x}^*) - A_k g_2(\mathbf{x}_k))$ on both sides to obtain,

$$\begin{aligned}
&A_{k+1}(g(\mathbf{x}_{k+1}) - g(\mathbf{x}^*)) + A_k(g(\mathbf{x}^*) - g(\mathbf{x}_k)) \\
&\leq a_k g_k - a_k g(\mathbf{x}^*) + A_{k+1}g_2(\mathbf{x}_{k+1}) - A_k g_2(\mathbf{x}_k) - a_k g_2(\mathbf{z}_{k+1}) + \frac{L_g a_k^2}{2A_{k+1}}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2
\end{aligned} \tag{79}$$

By the convexity of g_2, $A_{k+1} = A_k + a_k$, and $\mathbf{x}_{k+1} = \frac{A_k}{A_k + a_k}\mathbf{x}_k + \frac{a_k}{A_k + a_k}\mathbf{z}_{k+1}$ (outlined in Algorithm 2), i.e. $A_{k+1}g_2(\mathbf{x}_{k+1}) - A_k g_2(\mathbf{x}_k) - a_k g_2(\mathbf{z}_{k+1}) \leq 0$, the second inequality of this Lemma can be achieved. □

Hence, with the additional Assumption B.1, by replicating the analysis outlined in Section 4, we can derive identical complexity results for Algorithm 2 in either the compact domain setting or with the Hölderian error bounds on g.

C Connection with the Polyak Step Size

In this section, we would like to highlight the connection between our algorithm's projection step (outlined in Step 6 of Algorithm 1) and the Polyak step size. To make this connection, we first without loss of generality, replace g_k with g^*. It is a reasonable argument, as g_k values are close to g^*, a point highlighted in (8). In addition, we further assume that the set $\mathcal{Z} = \mathbb{R}^n$ to simplify the

expressions. Given these substitutions, the projection step in our AGM-BiO method is equivalent to solving the following problem:

$$\begin{aligned} \min \quad & \|\mathbf{x} - \mathbf{x}_k\|^2 \\ \text{s.t.} \quad & g(\mathbf{x}_k) + \langle \nabla g(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle \leq g^* \end{aligned}$$

In other words,$\mathbf{x}_{k+1}$ is the unique solution of the above quadratic program with a linear constraint. By writing the optimality conditions for the above problem and considering λ as the Lagrange multipliers associated with the linear constraint, we obtain that

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k - \lambda \nabla g(\mathbf{x}_k) \\ \lambda(g(\mathbf{x}_k) + \langle \nabla g(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle - g^*) = 0 \\ \lambda \geq 0 \end{cases}$$

Given the fact that $\mathbf{x}_{k+1} \neq \mathbf{x}_k$, we can conclude that $\lambda \neq 0$, and hence we have

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k - \lambda \nabla g(\mathbf{x}_k) \\ g(\mathbf{x}_k) + \langle \nabla g(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle - g^* = 0 \\ \lambda > 0 \end{cases}$$

By replacing $\mathbf{x}_{k+1}$ in the second expression with its expression in the first equation we obtain that

$$\lambda = \frac{g(\mathbf{x}_k) - g^*}{\|\nabla g(\mathbf{x}_k)\|^2}.$$

which is exactly the Polyak step size in the literature [Pol87]. To solve a bilevel optimization problem, we intend to do gradient descent for both upper- and lower-level functions. Tuning the ratio of upper- and lower-level step size is generally hard. However, by connecting the projection step with the Polyak step size, we observe that the stepsize for the lower-level objective is auto-selected as the Polyak stepsize in our method. In other words, it is one of the advantages of our algorithm that we do not need to choose the lower-level stepsize or ratio of the upper- and lower-level stepsize theoretically or empirically.

D Experiment Details

In this section, we include more details of the numerical experiments in Section 5. All simulations are implemented using MATLAB R2022a on a PC running macOS Sonoma with an Apple M1 Pro chip and 16GB Memory.

D.1 Over-parametrized Regression

Dataset generation. The original Wikipedia Math Essential dataset [Roz+21] composes of a data matrix of size 1068×731. We randomly select one of the columns as the outcome vector $\mathbf{b} \in \mathbb{R}^{1068}$ and the rest to be a new matrix $\mathbf{A} \in \mathbb{R}^{1068 \times 730}$. We set the constraint parameter $\lambda = 1$ in this experiment, i.e., the constraint set is given by $\mathcal{Z} = \{\boldsymbol{\beta} \mid \|\boldsymbol{\beta}\|_2 \leq 1\}$.

Implementation details. To be fair, all the algorithms start from the origin as the initial point. For our AGM-BiO method, we set the target tolerances for the absolute suboptimality

and infeasibility to $\epsilon_f = 10^{-4}$ and $\epsilon_g = 10^{-4}$, respectively. We choose the stepsizes as $a_k = 10^{-2}(k+1)/(4L_f)$. In each iteration, we need to do a projection onto an intersection of a L_2-ball and a halfspace, which has a closed-form solution. For a-IRG, we set $\eta_0 = 10^{-3}$ and $\gamma_0 = 10^{-3}$. For CG-BiO, we obtain an initial point with FW gap of the lower-level problem less than $\epsilon_g/2 = 5\times 10^{-5}$ and choose stepsize $\gamma_k = 10^{-2}/(k+2)$. For Bi-SG, we set $\eta_k = 10^{-2}/(k+1)^{0.75}$ and $t_k = 1/L_g = 1/\lambda_{max}(\mathbf{A}_{tr}^\top \mathbf{A}_{tr}) = 1.5 \times 10^{-4}$. For SEA, we set both the lower- and upper-level stepsizes to be 10^{-4}. For R-APM, since the lower-level problem does not satisfy the weak sharpness condition, we set $\eta = 1/(K+1) = 1.25 \times 10^{-5}$ and $\gamma = 10^{-4} \leq 1/(L_g + \eta L_f)$. For PB-APG, we set the penalty parameter $\gamma = 10^4$. Note that the lower-level problem in this experiment does not satisfy Höderian error bound assumption, so there is no theoretical guarantee for PB-APG.

D.2 Linear inverse problems

Dataset generation. We set $\mathbf{Q} = \mathbf{I}_n$, $\mathbf{A} = \mathbf{1}_n^\top$, and $\mathbf{b} = 1$. The constraint set is selected as $\mathcal{Z} = \mathbb{R}_+^n$. We choose a low dimensional ($n = 3$) and a high dimensional ($n = 100$) example and run $K = 10^3$ number of iterations to compare the numerical performance of these algorithms, respectively.

Implementation details. To be fair, all the algorithms start from the same initial point randomly chosen from $\mathbb{R}_+^n$. For our AGM-BiO method, we set the stepsizes as $a_k = \gamma(k+1)/(4L_f)$, where $\gamma = 1/(\frac{2L_g}{L_f}K^{2/3} + 2)$ as suggested in Theorem 4.4. In each iteration, we need to project onto an intersection of a halfspace and $\mathbb{R}_+^n$. Since halfspaces and $\mathbb{R}_+^n$ are both convex and closed set, the projection subproblem can be solved by Dykstra's projection algorithm in [Bau+11]. For a-IRG, we set $\eta_0 = 10^{-2}$ and $\gamma_0 = 10^{-2}$. For Bi-SG, we set $\eta_k = 1/(k+1)^{0.75}$ and $t_k = 1/L_g$. For SEA, we set the lower-level stepsize to be 10^{-2} and the upper-level stepsize to be 10^{-2}. For R-APM, since the lower-level problem does not satisfy the weak sharpness condition, we set $\eta = 1/(K+1)$ and $\gamma = 1/(L_g + \eta L_f)$. For PB-APG, we set the penalty parameter $\gamma = 10^4$. For Bisec-BiO, we choose the target tolerances to $\epsilon_f = \epsilon_g = 10^{-4}$. For comparison purposes, we limit the maximum number of gradient evaluations for each APG call to 10^2. In this experiment, $L_f = 1$ and $L_g = n$, where n is the number of dimensions.