Near-Optimal Convex Simple Bilevel Optimization with a Bisection Method

Jiulin Wang **Xu Shi** **Rujun Jiang**[†]
School of Data Science, Fudan University

Abstract

This paper studies a class of simple bilevel optimization problems where we minimize a composite convex function at the upper-level subject to a composite convex lower-level problem. Existing methods either provide asymptotic guarantees for the upper-level objective or attain slow sublinear convergence rates. We propose a bisection algorithm to find a solution that is ϵ_f-optimal for the upper-level objective and ϵ_g-optimal for the lower-level objective. In each iteration, the binary search narrows the interval by assessing inequality system feasibility. Under mild conditions, the total operation complexity of our method is $\tilde{\mathcal{O}}\left(\max\{\sqrt{L_{f_1}/\epsilon_f}, \sqrt{L_{g_1}/\epsilon_g}\}\right)$. Here, a unit operation can be a function evaluation, gradient evaluation, or the invocation of the proximal mapping, L_{f_1} and L_{g_1} are the Lipschitz constants of the upper- and lower-level objectives' smooth components, and $\tilde{\mathcal{O}}$ hides logarithmic terms. Our approach achieves a near-optimal rate, matching the optimal rate in unconstrained smooth or composite convex optimization when disregarding logarithmic terms. Numerical experiments demonstrate the effectiveness of our method.

1 INTRODUCTION

In this paper, we focus on the following convex bilevel optimization problem:

$$\text{(P)} \quad \begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}) := f_1(\mathbf{x}) + f_2(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \arg\min_{\mathbf{z}\in\mathbb{R}^n} g(\mathbf{z}) := g_1(\mathbf{z}) + g_2(\mathbf{z}). \end{aligned} \tag{1}$$

Proceedings of the 27[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. [†]Corresponding Author.

Here, functions f_1 and $g_1 : X \to \mathbb{R}$ are convex and continuously differentiable over an open set $X \in \mathbb{R}^n$. Their gradients, ∇f_1 and ∇g_1, are L_{f_1}- and L_{g_1}-Lipschitz continuous, respectively. f_2 and $g_2 : \mathbb{R}^n \to \mathbb{R} \cup \infty$ are proper lower semicontinuous (l.s.c.) convex functions. We assume that g is not strongly convex, and the lower-level problem has multiple optimal solutions; in other words, the optimal solution set of the lower-level problem, denoted as X_g^*, is not a singleton. Otherwise, the optimal minimum is determined by the lower-level problem.

This specific class of problems, often known as "simple bilevel optimization" in the existing literature (Dempe et al., 2010; Dutta and Pandit, 2020; Shehu et al., 2021; Jiang et al., 2023), is a subclass of the general bilevel optimization problems. In a general bilevel optimization, the lower-level problem is parametrized by some upper-level variables. Bilevel optimization has garnered significant interest owing to its versatile applications across domains such as reinforcement learning (Hong et al., 2020), meta-learning (Bertinetto et al., 2018; Rajeswaran et al., 2019), hyper-parameter optimization (Franceschi et al., 2018; Shaban et al., 2019), and adversarial learning (Bishop et al., 2020; Wang et al., 2021, 2022).

Let p^* be the optimal value of problem (1) and g^* be the optimal value of the unconstrained lower-level problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; g(\mathbf{x}) := g_1(\mathbf{x}) + g_2(\mathbf{x}). \tag{2}$$

The goal of this paper is to find an (ϵ_f, ϵ_g)-optimal solution $\hat{\mathbf{x}}$ satisfying

$$f(\hat{\mathbf{x}}) - p^* \le \epsilon_f \quad \text{and} \quad g(\hat{\mathbf{x}}) - g^* \le \epsilon_g.$$

A possible approach for solving problem (1) is to reformulate it to a constrained optimization problem with functional constraints and apply primal-dual methods. Specifically, problem (1) can be reformulated as a constrained convex optimization problem as follows:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \;\; \text{s.t.} \;\; g(\mathbf{x}) \le g^*. \tag{3}$$

A critical issue of applying primal-dual-type methods is that problem (3) does not satisfy the regularity condition required for their convergence (the strict feasibility does not hold and hence Slater's condition fails). Furthermore, classical first-order algorithms, such as projected gradient descent, may also be ineffective due to the difficulty of computing the orthogonal projection onto the level-set of the lower-level objective. If we relax the constraint and solve the following problem to ensure strict feasibility

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \ \text{ s.t. } \ g(\mathbf{x}) \le g^* + \epsilon, \tag{4}$$

these challenges remain. Indeed, as ϵ approaches zero, causing the problem to become nearly degenerate, the dual optimal variable may tend towards infinity. This phenomenon hinders convergence and results in numerical instability (Bonnans and Shapiro, 2013). Consequently, problem (1) cannot be straightforwardly addressed as a conventional constrained optimization problem; instead, it necessitates novel theories and algorithms customized for its hierarchical structure.

1.1 Our Approach

Our main technique is a bisection method that iteratively narrows an interval $[l, u]$ that includes p^*. The binary search is based on the feasibility of the following system:

$$f(\mathbf{x}) \le c, \ g(\mathbf{x}) \le g^*, \tag{5}$$

where $c = \frac{l+u}{2}$. The key observation is that if System (5) is feasible, then c is an upper bound of p^*; otherwise, System (5) is infeasible and c is a lower bound of p^*. This process divides the interval in half. The feasibility of the system can be checked by solving the following problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \ g(\mathbf{x}), \ \ \text{s.t.} \ \ f(\mathbf{x}) \le c. \tag{6}$$

The above is only a basic idea, the detail of our algorithm that considers the inexactness of solving (6) is detailed in Section 3. Moreover, by showing that each iteration and the initial lower and upper bounds can be solved by Accelerated Proximal Gradient (APG) methods (Nesterov, 1983; Beck and Teboulle, 2009), we derive a comprehensive complexity analysis for our algorithm.

We state our contributions in the following:

- Under mild conditions, we propose a novel bisection method that finds an (ϵ_f, ϵ_g)-optimal solution of problem (1) with an operation complexity $\tilde{\mathcal{O}}\left(\max\{\sqrt{L_{f_1}/\epsilon_f}, \sqrt{L_{g_1}/\epsilon_g}\}\right)$, where the notation $\tilde{\mathcal{O}}$ suppresses a logarithmic term. Our method achieves near-optimal non-asymptotic guarantees on both upper- and lower-level problems, i.e., our rate aligns with the optimal rate observed in unconstrained smooth or composite convex optimization, with the exception of omitting the logarithmic term (Nemirovsky and Yudin, 1983; Woodworth and Srebro, 2016).
- With an additional r-th-order ($r \ge 1$) Hölderian error bound assumption on the lower-level problem and incorporating other smoothness assumptions, our method can find a solution $\hat{\mathbf{x}}$ that satisfies $|f(\hat{\mathbf{x}}) - p^*| \le \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \le \epsilon_g$ with an $\tilde{\mathcal{O}}\left(1/\sqrt{\epsilon_f^r}\right)$ operation complexity. This complexity arises under the setting $\epsilon_g = \frac{\alpha}{\gamma}\left(\frac{\epsilon_f}{B_f}\right)^r$, where α, γ, B_f are defined in Section 4.1.
- With an additional assumption on the optimal value of (4), our method can find a solution $\hat{\mathbf{x}}$ that satisfies $|f(\hat{\mathbf{x}}) - p^*| \le \epsilon_f$ and $g(\hat{\mathbf{x}}) - g^* \le \epsilon_g$ with an $\tilde{\mathcal{O}}\left(\max\{1/\sqrt{L_{\epsilon_g}\epsilon_g}, 1/\sqrt{\epsilon_g}\}\right)$ operation complexity. This operational complexity is observed when $\epsilon_f = L_{\epsilon_g}\epsilon_g$, where L_{ϵ_g} is defined in Section 4.2.
- Numerical experiment results on different problems demonstrate the superior performance of our method compared to the state-of-the-art.

1.2 Related Work

One class of algorithms to solve problem (1) is based on solving the Tikhonov-type regularization (Tikhonov and Arsenin, 1977):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := g(\mathbf{x}) + \lambda f(\mathbf{x}),$$

where $\lambda > 0$ is a regularization parameter. However, these kinds of algorithms fail to provide any non-asymptotic guarantee for either the upper- or lower-level objective. For a review of these algorithms, see Doron and Shtern (2023) and Jiang et al. (2023).

Another class of algorithms aims to establish non-asymptotic convergence rates for problem (1). Beck and Sabach (2014) presented the Minimal Norm Gradient (MNG) method for the case where f is strongly convex. They demonstrated that MNG converges asymptotically to the optimal solution and possesses an $\mathcal{O}\left(L_{g_1}^2/\epsilon^2\right)$ complexity bound for the lower-level problem. In their setting, $g \equiv g_1$ and $g_2 \equiv 0$. Developed from the sequential averaging method (SAM) framework, the Bilevel Gradient Sequential Averaging Method (BiG-SAM) is proposed by Sabach and Shtern (2017). This algorithm can achieve an $\mathcal{O}\left(L_{g_1}/\epsilon\right)$ complexity bound for the lower-level problem. Solodov (2007) introduced the Iterative Regularized Projected

Table 1: Summary of simple bilevel optimization algorithms. The abbreviations "SC", "C", "C3" stand for "strongly convex", "convex", "Convex objective with Convex Compact constraints" respectively. When the connection between complexity and the gradient's Lipschitz constant is clear, we include it in the complexity result; otherwise, we omit it.

References	Upper-level	Lower-level	Convergence	
	Objective f	Objective g	Upper-level	Lower-level
MNG (Beck and Sabach, 2014).	SC, differentiable	C, smooth	Asymptotic	$\mathcal{O}\left(L_{g_1}^2/\epsilon^2\right)$
BiG-SAM (Sabach and Shtern, 2017)	SC, smooth	C, composite	Asymptotic	$\mathcal{O}\left(L_{g_1}/\epsilon\right)$
IR-IG (Amini and Yousefian, 2019)	SC	C3, Finite sum	Asymptotic	$\mathcal{O}\left(1/\epsilon^4\right)$
Tseng's method (Malitsky, 2017)	C, composite	C, composite	Asymptotic	$\mathcal{O}\left(1/\epsilon\right)$
ITALEX (Doron and Shtern, 2023)	C, composite	C, composite	$\mathcal{O}\left(1/\epsilon^2\right)$	$\mathcal{O}\left(1/\epsilon\right)$
a-IRG (Kaushik and Yousefian, 2021)	C, Lipschitz	C, Lipschitz	$\mathcal{O}\left(\max\{1/\epsilon_f^4, 1/\epsilon_g^4\}\right)$	
CG-BiO (Jiang et al., 2023)	C, smooth	C3, smooth	$\mathcal{O}\left(\max\{L_{f_1}/\epsilon_f, L_{g_1}/\epsilon_g\}\right)$	
Our method	C, composite	C, composite	$\tilde{\mathcal{O}}\left(\max\{\sqrt{L_{f_1}/\epsilon_f}, \sqrt{L_{g_1}/\epsilon_g}\}\right)$	

Gradient (IR-PG) method, which involves applying a projected gradient step to the Tikhonov-type regularization function $\phi(\mathbf{x})$ at each iteration. Amini and Yousefian (2019) extended the IR-PG method (Solodov, 2007) for the case where f is strongly convex but not necessarily differentiable. Their method achieves a convergence rate of $\mathcal{O}\left(1/k^{0.5-b}\right)$ for the lower-level problem, where $b \in (0, 0.5)$. Malitsky (2017) studied a version of Tseng's accelerated gradient method that obtains a convergence rate of $\mathcal{O}\left(1/k\right)$ for the lower-level problem. These prior works only establish the convergence rate for the lower-level problem, while the rate for the upper-level objective is missing.

Several algorithms have recently provided convergence rates for both upper- and lower-level objectives. Doron and Shtern (2023) presented a scheme called Iterative Approximation and Level-set Expansion (ITALEX) to solve problem (1). Their algorithm achieves convergence rates of $\mathcal{O}\left(1/k\right)$ and $\mathcal{O}\left(1/\sqrt{k}\right)$ for the lower- and upper-level problems, respectively. Kaushik and Yousefian (2021) showed that an iteratively regularized gradient (a-IRG) method can obtain complexity $\mathcal{O}\left(1/k^{0.5-b}\right)$ for the upper-level problem and $\mathcal{O}\left(1/k^b\right)$ for the lower-level, where $b \in (0, 0.5)$. To balance the two rates, one can set $b = 0.25$, and the complexity bound is $\mathcal{O}\left(\max\{1/\epsilon_f^4, 1/\epsilon_g^4\}\right)$ as stated in Table 1. Jiang et al. (2023) presented a conditional gradient-based bilevel optimization (CG-BiO) method, which requires $\mathcal{O}\left(\max\{L_{f_1}/\epsilon_f, L_{g_1}/\epsilon_g\}\right)$ operation complexity to find an (ϵ_f, ϵ_g)-optimal solution. In their setting, $f \equiv f_1$ and $f_2 \equiv 0$.

2 PRELIMINARIES

In this paper, we use an Accelerated Proximal Gradient (APG) method to approximately solve subproblems, which have the following form:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \ \varphi(\mathbf{x}) := \varphi_1(\mathbf{x}) + \varphi_2(\mathbf{x}), \tag{7}$$

where the function $\varphi_1 : X \to \mathbb{R}$ is convex and continuously differentiable on an open set $X \in \mathbb{R}^n$. The gradient $\nabla\varphi_1$ is L_{φ_1}-Lipschitz continuous. The function $\varphi_2 : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is proper, lower semicontinuous, convex, possibly non-smooth, and proximal-friendly. A function h is proximal-friendly means that the proximal mapping of h, defined as

$$\text{prox}_h(\mathbf{y}) = \arg\min_{\mathbf{x}\in\mathbb{R}^n} h(\mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2,$$

is easy to compute. In this paper, we take the classical Fast Iterative Shrinkage Thresholding Algorithm (FISTA) proposed in Beck and Teboulle (2009) as an APG algorithm (see more details in the appendix). Next, we give a definition of an APG oracle.

Definition 1. *Given φ_1, φ_2, L_{φ_1}, and $\mathbf{x}_0^\varphi \in \mathbb{R}^n$ as defined above, an APG oracle, denoted by $\tilde{\mathbf{x}}^\varphi = APG(\varphi_1, \varphi_2, L_{\varphi_1}, \mathbf{x}_0^\varphi, \epsilon)$, is a procedure that implements the classical FISTA scheme within $\mathcal{O}(\sqrt{L_{\varphi_1}/\epsilon})$ iterations to obtain an ϵ-optimal solution to problem (7), denoted as $\tilde{\mathbf{x}}_\varphi$.*

If the Lipschitz constant L_{φ_1} is unknown or computationally infeasible, we can apply the FISTA scheme with line search as an alternative to the APG algorithm (see the appendix).

For any fixed c, problem (6) can be rewritten in the following form:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \ g_c(\mathbf{x}) := g_1(\mathbf{x}) + h_c(\mathbf{x}), \tag{8}$$

where $h_c(\mathbf{x}) := g_2(\mathbf{x}) + \delta_c(\mathbf{x})$ and $\delta_c(\mathbf{x})$ is an indicator function with the definition that $\delta_c(\mathbf{x}) = 0$ if $f(\mathbf{x}) \leq c$; $\delta_c(\mathbf{x}) = +\infty$ if $f(\mathbf{x}) > c$. We provide some motivating examples in which the proximal mapping of h_c is easy to compute. These examples can be found in the appendix.

2.1 Assumptions

Assumption 1. *We adopt the following basic assumptions.*

(i) Functions f_1 and g_1 are convex and continuously differentiable. The gradients of the functions f_1, g_1, denoted by ∇f_1 and ∇g_1 are L_{f_1}- and L_{g_1}-Lipschitz continuous, respectively.

(ii) Functions f_2 and g_2 are proper, lower semicontinuous, convex, possibly non-smooth, and proximal-friendly.

(iii) The upper- and lower-level functions are lower bounded:

$$f^* := \inf_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) > -\infty,\ g^* := \inf_{\mathbf{x}\in\mathbb{R}^n} g(\mathbf{x}) > -\infty.$$

In addition, we assume that g is not strongly convex, and the lower-level problem has multiple optimal solutions.

(iv) For any fixed c, the function $h_c := g_2 + \delta_c$ in problem (8) is proximal-friendly.

Remark 1. *We assume that the upper-level problem involves the minimization of a composite convex function, which comprises the sum of a smooth convex function and a potentially non-smooth convex function. This assumption is significantly weaker than the strong-convexity assumption made in certain previous studies (Beck and Sabach, 2014; Sabach and Shtern, 2017; Amini and Yousefian, 2019). This assumption is also less restrictive than the requirement for the upper-level objective function to be smooth (Jiang et al., 2023). We also assume that the lower-level problem is a composite convex minimization. This assumption is less restrictive than the smoothness assumption made in Beck and Sabach (2014). Additionally, this assumption is weaker than the requirement that the lower-level objective function is convex with convex compact constraints, as described in Amini and Yousefian (2019) and Jiang et al. (2023).*

Remark 2. *We make Assumption 1(iv) to enable the efficient application of the APG oracle for solving problem (6). The function h_c is the sum of two convex functions, and the study of proximal mapping for such sums can be found in the literature (Yu, 2013; Pustelnik and Condat, 2017; Bauschke et al., 2018; Adly et al., 2019).*

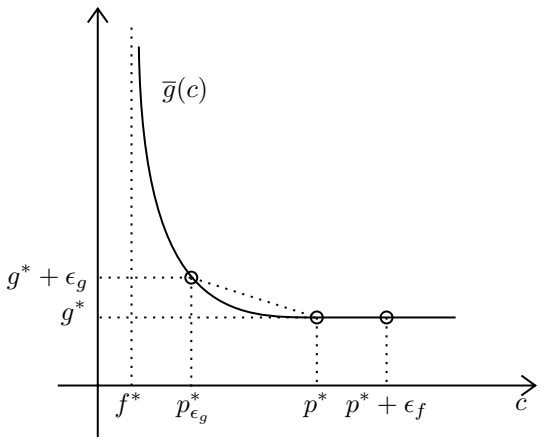

Figure 1: Variation of $\bar{g}(c)$ over $(f^*, +\infty)$

3 MAIN ALGORITHM AND CONVERGENCE ANALYSIS

Before the presentation of the main algorithm, we describe our idea in the next subsection.

3.1 Bisection Method

Our method is a bisection method whose heart is the feasibility of System (5). Let f^* be the optimal value of the unconstrained upper-level problem

$$\min_{\mathbf{x}\in\mathbb{R}^n}\ f(\mathbf{x}) := f_1(\mathbf{x}) + f_2(\mathbf{x}). \tag{9}$$

For a given $c > f^*$, we let $\bar{g}(c)$ be the optimal value of problem (6). Then $\bar{g}(c)$ is a univariate function of c on $(f^*, +\infty)$. According to Theorem 5.3 in Rockafellar (1970), the function $\bar{g}(c)$ is convex. The function $\bar{g}(c)$ is also non-increasing as the feasible set of problem (6) becomes larger when c increases. Moreover, if $f^* < c < p^*$, then the inequality $\bar{g}(c) > g^*$ holds; otherwise $c \geq p^*$ and we have $\bar{g}(c) = g^*$. Therefore, p^* is the left-most root of the equation $\bar{g}(c) = g^*$. Let $p^*_{\epsilon_g}$ be the optimal value of (4) with $\epsilon = \epsilon_g$, then it is a root of the equation $\bar{g}(c) = g^* + \epsilon_g$. We illustrate the graph of $\bar{g}(c)$ in Figure 1.

To illustrate the basic idea of our method, we make an ideal assumption that the exact values of g^* and $\bar{g}(c)$ can be obtained. We can observe that if $\bar{g}(c) > g^*$ holds, then System (5) is infeasible; otherwise, $\bar{g}(c) = g^*$ and System (5) is feasible. For a guess point c, if the condition $\bar{g}(c) > g^*$ holds, then c is a lower bound of p^*; otherwise, c is an upper bound of p^*.

However, the ideal assumption that the exact values of g^* and $\bar{g}(c)$ can be obtained does not hold. Instead, we solve problem (2) and problem (6) to approximate them, respectively. For problem (2), we invoke the APG oracle $\tilde{\mathbf{x}}_g$=APG$(g_1, g_2, L_{g_1}, \mathbf{x}_0^g, \epsilon_g/2)$ to solve it.

Then we can find an approximate solution $\tilde{\mathbf{x}}_g$ that satisfies

$$0 \leq \tilde{g} - g^* \leq \epsilon_g/2, \tag{10}$$

where $\tilde{g} = g(\tilde{\mathbf{x}}_g)$.

Under Assumption 1(iv), the proximal mapping of h_c is easy to compute. Then we can apply an APG oracle to problem (6). For a given c, we invoke the APG oracle $\tilde{\mathbf{x}}_c$=APG$(g_1, h_c, L_{g_1}, \mathbf{x}_0^c, \epsilon_g/2)$ to solve problem (6). Then we can obtain an approximate solution $\tilde{\mathbf{x}}_c$ that satisfies

$$0 \leq g(\tilde{\mathbf{x}}_c) - \bar{g}(c) \leq \epsilon_g/2. \tag{11}$$

Since the condition $\bar{g}(c) > g^*$ cannot be verified directly, we replace it with the following verifiable condition

$$g(\tilde{\mathbf{x}}_c) > \tilde{g} + \epsilon_g/2. \tag{12}$$

If Condition (12) holds, then we have

$$\bar{g}(c) \overset{(11)}{\geq} g(\tilde{\mathbf{x}}_c) - \epsilon_g/2 \overset{(12)}{>} \tilde{g} \overset{(10)}{\geq} g^*,$$

i.e., the inequality $\bar{g}(c) > g^*$ holds. Thus, System (5) is infeasible, and c is a lower bound of p^*.

If Condition (12) does not hold, we have $g(\tilde{\mathbf{x}}_c) \leq \tilde{g} + \epsilon_g/2$ and thus

$$\bar{g}(c) \overset{(11)}{\leq} g(\tilde{\mathbf{x}}_c) \leq \tilde{g} + \epsilon_g/2 \overset{(10)}{\leq} g^* + \epsilon_g. \tag{13}$$

We cannot infer that $\bar{g}(c) = g^*$ holds in this case. Also, we cannot claim that System (5) is feasible. Thus c might not be an upper bound of p^*. By (13), $\tilde{\mathbf{x}}_c$ is a feasible solution of (4) with $\epsilon = \epsilon_g$. Therefore, $f(\tilde{\mathbf{x}}_c)$ is an upper bound on $p^*_{\epsilon_g}$, where $p^*_{\epsilon_g}$ represents the optimal value of (4) with $\epsilon = \epsilon_g$ as previously defined. In addition, as the inequality $g(\tilde{\mathbf{x}}_c) \leq g^* + \epsilon_g$ holds, $\tilde{\mathbf{x}}_c$ is an ϵ_g-optimal solution of the lower-level problem (2).

Summarizing the above analysis, we present the following lemma.

Lemma 1. *For any fixed c, if Condition (12) is satisfied, then System (5) is infeasible, and c is a lower bound of p^*. If Condition (12) is not satisfied, then we can obtain $\tilde{\mathbf{x}}_c$ as an ϵ_g-optimal solution of the lower-level problem and $f(\tilde{\mathbf{x}}_c)$ is an upper bound of $p^*_{\epsilon_g}$.*

Next, we show how to obtain the initial interval $[l, u]$ for the bisection procedure. Here l is a lower bound of p^* and u is an upper bound of $p^*_{\epsilon_g}$, but possibly not an upper bound of p^*. We invoke the APG oracle $\tilde{\mathbf{x}}_f$=APG$(f_1, f_2, L_{f_1}, \mathbf{x}_0^f, \epsilon_f/2)$ to solve problem (9). Then we can obtain an approximate solution $\tilde{\mathbf{x}}_f$ that satisfies

$$0 \leq f(\tilde{\mathbf{x}}_f) - f^* \leq \epsilon_f/2. \tag{14}$$

We have $f(\tilde{\mathbf{x}}_f) - \epsilon_f/2 \leq f^* \leq p^*$. We use $l = f(\tilde{\mathbf{x}}_f) - \epsilon_f/2$ as an initial lower bound of p^*. Moreover, since $\tilde{\mathbf{x}}_g$ satisfies (10), we have $\tilde{\mathbf{x}}_g$ is a feasible solution of (4) with $\epsilon = \epsilon_g$. Then we use $u = f(\tilde{\mathbf{x}}_g)$ as an initial upper bound of $p^*_{\epsilon_g}$.

Now we can do a binary search over $[l, u]$. For a given $c = \frac{l+u}{2}$, we check whether Condition (12) is satisfied. If Condition (12) is satisfied, we let $l = c$ be a new lower bound of p^*. If Condition (12) is not satisfied, we let $u = f(\tilde{\mathbf{x}}_c)$, which is less than or equal to c, be a new upper bound of $p^*_{\epsilon_g}$. We summarise our method in Algorithm 1.

Algorithm 1 Bisection-based method for simple Bilevel Optimization (Bisec-BiO)

Input: $f_1, f_2, g_1, g_2, L_{f_1}, L_{g_1}, \epsilon_f, \epsilon_g$
Output: An (ϵ_f, ϵ_g)-optimal solution $\hat{\mathbf{x}}$

```
Invoke APG oracles to obtain initial bounds l and u, and the approximate solutions x̃_f and x̃_g.
while u − l > ε_f do
    let c = (l+u)/2 and invoke an APG oracle to obtain an approximate solution x̃_c.
    if Condition (12) is satisfied then
        let l = c,
    else
        let u = f(x̃_c).                    ▷ f(x̃_c) ≤ c
    end if
end while
Let c = u and return the corresponding x̃_c as x̂.
```

3.2 Convergence Analysis under Assumption 1

In this subsection, we give the complexity result of our method.

Theorem 1. *Suppose Assumption 1 holds. Algorithm 1 produces an (ϵ_f, ϵ_g)-optimal solution for problem (1) after at most T evaluations of the function values f_1, f_2, g_1 and g_2, the gradients ∇f_1 and ∇g_1, and the calls of proximal mapping with respect to function h_c, where*

$$T = \tilde{\mathcal{O}}\left(\max\left\{\sqrt{\frac{L_{f_1}}{\epsilon_f}}, \sqrt{\frac{L_{g_1}}{\epsilon_g}}\right\}\right),$$

and $\tilde{\mathcal{O}}$ suppresses a logarithmic term.

Theorem 1 demonstrates that our complexity achieves the near-optimal rate for both upper- and lower-level objectives and matches the optimal rate of first-order methods for unconstrained smooth or composite convex optimization when disregarding the logarithmic term (Nemirovsky and Yudin, 1983; Woodworth and Srebro, 2016). Comparing with existing works (Beck

and Sabach, 2014; Sabach and Shtern, 2017; Amini and Yousefian, 2019; Malitsky, 2017; Doron and Shtern, 2023; Kaushik and Yousefian, 2021; Jiang et al., 2023), our result provides the best-known non-asymptotic bounds for both upper- and lower-level objectives. Specifically, our complexity bound improves upon the result by Jiang et al. (2023) by orders of magnitude. They considered a different setup where the upper-level function is smooth, and the lower-level objective is a smooth convex function with convex compact constraints.

Remark 3. *Assumption 1(iv) essentially implies that projecting onto the sublevel set of f is straightforward, as observed in the motivating examples. Consequently, we can employ a bisection method to locate f^* by verifying the solvability of the projection onto the sublevel set of f. This approach eliminates the dependency on L_{f_1} from the overall complexity.*

4 CONVERGENCE ANALYSIS UNDER OTHER ASSUMPTIONS

In this section, we provide an analysis of the metric $f(\hat{\mathbf{x}}) - p^*$. Our method guarantees that $f(\hat{\mathbf{x}})$ serves as an upper bound for $p^*_{\epsilon_g}$; however, it may not necessarily be an upper bound for p^*, which could result in a negative value for $f(\hat{\mathbf{x}}) - p^*$. In fact, as $\hat{\mathbf{x}}$ may not be an exact optimal solution to the lower-level problem, it may not be a feasible point for problem (1). Hence, the value of $f(\hat{\mathbf{x}}) - p^*$ may be negative. In essence, we currently offer an upper bound metric for $f(\hat{\mathbf{x}}) - p^*$ while lacking a corresponding lower bound metric. Although this may appear a bit unconventional, it is noteworthy that Kaushik and Yousefian (2021) and Jiang et al. (2023) similarly employed $f(\hat{\mathbf{x}}) - p^*$ as their performance metric.

In this section, we introduce additional assumptions to establish a lower bound for $f(\hat{\mathbf{x}}) - p^*$, allowing us to provide a metric expressed as $|f(\hat{\mathbf{x}}) - p^*|$.

4.1 Convergence Analysis under Hölderian Error Bound Assumption

In this subsection, we make some additional assumptions to provide a lower bound for $f(\hat{\mathbf{x}}) - p^*$.

Assumption 2. *(i) The domain of g_2 is bounded.*

(ii) The function f_2 is l_{f_2}-Lipschitz continuous on $\mathrm{dom}(g_2)$, i.e. $|f_2(\mathbf{x}) - f_2(\mathbf{y})| \leq l_{f_2}\|\mathbf{x} - \mathbf{y}\|$.

In the following, we give some remarks on Assumption 2. Assumption 2(i) is fulfilled by many examples, see Section 5 in Amini and Yousefian (2019) and Section 2 in Jiang et al. (2023). In particular, in Section 5.2, we have $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \leq \lambda\}$, then $\mathrm{dom}(g_2)$ is bounded. As f_1 is continuous, ∇f_1 is bounded on $\mathrm{dom}(g_2)$. Hence Assumption 2(i) implies

$$B_{f_1} = \max_{\mathbf{x} \in \mathrm{dom}(g_2)} \|\nabla f_1(\mathbf{x})\|.$$

Then by mean-value theorem, the function f_1 is B_{f_1}-Lipschitz continuous on $\mathrm{dom}(g_2)$.

Assumption 2(ii) is mild. For example, if $f_2(\mathbf{x}) = \|\mathbf{x}\|_1$, then $l_{f_2} = \sqrt{n}$, where n is the dimension of $\mathbf{x}$. Let $B_f = B_{f_1} + l_{f_2}$. Under Assumption 2(i) and (ii), it follows that f is B_f-Lipschitz continuous on $\mathrm{dom}(g_2)$, namely,

$$|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq B_f\|\mathbf{x}_1 - \mathbf{x}_2\| \ \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathrm{dom}(g_2) \tag{15}$$

Assumption 3. *The function g satisfies the Hölderian error bound for some $\alpha > 0$ and $r \geq 1$ on the lower-level optimal solution set X^*_g, i.e,*

$$\frac{\alpha}{r}\ \mathrm{dist}(\mathbf{x}, X^*_g)^r \leq g(\mathbf{x}) - g^*,\ \forall \mathbf{x} \in \mathrm{dom}(g_2). \tag{16}$$

It is important to highlight that the error bound condition described in (16) has received considerable attention in the literature, as evidenced by studies such as Pang (1997); Bolte et al. (2017); Zhou and So (2017), and the associated references therein. Bolte et al. (2017) demonstrated that this error bound condition typically holds when the function g exhibits properties of being semi-algebraic and continuous, while also ensuring that $\mathrm{dom}(g_2)$ remains bounded. They also showed that there is an equivalence between Hölderian error bound condition and the Kurdyka-Łojasiewicz inequality. There are two notable special cases: (i) $r = 1$, X^*_g is a set of weak sharp minima of g (Burke and Deng, 2005); (ii) $r = 2$, Condition (16) is known as the quadratic growth condition (Drusvyatskiy and Lewis., 2018). Based on Corollary 5.1 in Li and Pong (2018) and Theorem 5 in Bolte et al. (2017), it is evident that in the motivating examples provided in the appendix, the lower-level objectives satisfy the Hölderian error bound assumption with $r = 2$.

Given Assumptions 2 and 3, we can derive the following lower bound for $f(\hat{\mathbf{x}}) - p^*$. Importantly, this result is an inherent characteristic of problem (1) and remains unaffected by the choice of algorithm.

Proposition 1. *Under Assumptions 2 and 3, let $\hat{\mathbf{x}}$ be an ϵ_g-optimal solution of the lower-level problem, i.e., $\hat{\mathbf{x}}$ satisfies $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$. Then it holds that:*

$$f(\hat{\mathbf{x}}) - p^* \geq -B_f \left(\frac{r\epsilon_g}{\alpha}\right)^{\frac{1}{r}}.$$

This result is similar to Proposition 1 in Jiang et al. (2023), which also gives a lower bound for $f(\hat{\mathbf{x}}) - p^*$ with similar assumptions. Combining Theorem 1 with Proposition 1, we have the following result.

Corollary 1. *Under Assumptions 1-3, let* $\hat{\mathbf{x}}$ *be the output of Algorithm 1 and set* $\epsilon_g = \frac{\alpha}{\gamma}\left(\frac{\epsilon_f}{B_f}\right)^r$. *Then with an operation complexity of* $\tilde{\mathcal{O}}\left(1/\sqrt{\epsilon_f^r}\right)$, *we can find an* $\hat{\mathbf{x}}$ *such that*

$$|f(\hat{\mathbf{x}}) - p^*| \leq \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g.$$

Corollary 1 illustrates that, under Assumptions 1-3, we can find an iteration point to be ϵ_f-close to optimality with an operation complexity $\tilde{\mathcal{O}}\left(1/\sqrt{\epsilon_f^r}\right)$.

4.2 Convergence Analysis under an Additional Assumption on the Optimal Value of (4)

In this subsection, we present an alternative approach to establishing a lower bound for $f(\hat{\mathbf{x}}) - p^*$. Denote the optimal value of problem (4) as $v(\epsilon)$. Here, $v(\epsilon)$ is a function of ϵ defined on the interval $[0, +\infty)$. According to the definitions of p^* and $p^*_{\epsilon_g}$, we can establish that $v(0) = p^*$ and $v(\epsilon_g) = p^*_{\epsilon_g}$. As illustrated in Figure 1, we can define an angle function $\theta(\epsilon)$ on $[0, \pi/2)$ such that

$$\epsilon = (v(0) - v(\epsilon)) \cdot \tan\theta(\epsilon).$$

Similar to the monotonicity of $\bar{g}(c)$, functions $v(\epsilon)$ and $\theta(\epsilon)$ are also monotonically non-decreasing. If $\epsilon = 0$, then $\theta(\epsilon) = 0$; otherwise $\epsilon > 0$ and $\theta(\epsilon) > 0$. It is hard to compute the exact value of $\tan\theta(\epsilon)$. Instead, for any fixed $\epsilon > 0$, we assume that there exists a parameter $L_\epsilon > 0$ such that $L_\epsilon \tan\theta(\epsilon) \geq 1$ holds. In other words, we make the following assumption.

Assumption 4. *For any fixed* $\epsilon > 0$, *we assume that there exists a parameter* $L_\epsilon > 0$ *such that* $v(0) - v(\epsilon) \leq L_\epsilon \epsilon$ *holds.*

In the following, we consider a simple two-dimensional toy example in which Assumption 4 holds.

Example 1 (A toy example)**.**

$$\begin{array}{ll} \min\limits_{\mathbf{x}\in\mathbb{R}^2} & f(\mathbf{x}) = |\mathbf{x}_1| + |\mathbf{x}_2| \\ \text{s.t.} & \mathbf{x} \in \arg\min\limits_{\mathbf{z}\in\mathbb{R}^2} g(\mathbf{z}) = (\mathbf{z}_1 - 1)^2. \end{array} \tag{17}$$

In this example, the optimal solution and the optimal value of (17) are $(\mathbf{x}_1^*, \mathbf{x}_2^*) = (1, 0)$ and $v(0) = p^* = 1$, respectively. Next, we consider problem (4) with $0 < \epsilon < 1$. It can be easily obtained that the optimal solution and the optimal value of (4) are $(\mathbf{x}_1^*, \mathbf{x}_2^*) = (1 - \sqrt{\epsilon}, 0)$ and $v(\epsilon) = 1 - \sqrt{\epsilon}$, respectively. Then we have

$$v(0) - v(\epsilon) = \sqrt{\epsilon}.$$

By setting $L_\epsilon \geq 1/\sqrt{\epsilon}$, Assumption 4 holds.

Under Assumption 4, we can derive a lower bound for $f(\hat{\mathbf{x}}) - p^*$ that is independent of the choice of algorithm.

Proposition 2. *Under Assumption 4, let* $\hat{\mathbf{x}}$ *be an* ϵ_g*-optimal solution of the lower-level problem, i.e.,* $\hat{\mathbf{x}}$ *satisfies* $g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g$. *Then it holds that:*

$$f(\hat{\mathbf{x}}) - p^* \geq -L_{\epsilon_g}\epsilon_g.$$

By combining Theorem 1 with Proposition 2, we have the following result.

Corollary 2. *Under Assumption 1 and Assumption 4, let* $\hat{\mathbf{x}}$ *be the output of Algorithm 1 and set* $\epsilon_f = L_{\epsilon_g}\epsilon_g$. *Then with an operation complexity of* $\tilde{\mathcal{O}}\left(\max\{1/\sqrt{L_{\epsilon_g}\epsilon_g}, 1/\sqrt{\epsilon_g}\}\right)$, *we can find an* $\hat{\mathbf{x}}$ *such that*

$$|f(\hat{\mathbf{x}}) - p^*| \leq \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g.$$

Corollary 2 demonstrates that with Assumptions 1 and 4 in place, we can obtain an iteration point to be $L_{\epsilon_g}\epsilon_g$-close to optimality with an operation complexity $\tilde{\mathcal{O}}\left(\max\{1/\sqrt{L_{\epsilon_g}\epsilon_g}, 1/\sqrt{\epsilon_g}\}\right)$.

5 NUMERICAL EXPERIMENTS

In this section, we apply our method (Bisec-BiO) to two bilevel optimization problems from the motivating examples in the appendix and compare its performance with other existing methods in the literature (Beck and Sabach, 2014; Sabach and Shtern, 2017; Kaushik and Yousefian, 2021; Gong and Liu, 2021; Jiang et al., 2023). For all experiments, we set $\epsilon_f = 10^{-5}$ and $\epsilon_g = 10^{-6}$, and we adopt the Greedy FISTA algorithm proposed in Liang et al. (2022) as the APG method. The Greedy FISTA algorithm can achieve superior practical performance compared to the classical FISTA.

5.1 Minimum Norm Solution Problem (MNP)

We first consider the linear regression problem on the YearPredictionMSD dataset[1], which contains information on 515,345 songs, with a release year from 1992 to 2011. For each song, the dataset contains its release year and an additional 90 attributes. We use a sample of 1,000 songs randomly selected from the dataset with uniform i.i.d distribution, and denote the feature matrix and the release years by A and b, respectively. Additionally, in line with Merchav and Sabach (2023), we adopt a min-max scaling technique and add an interceptor and 90 co-linear attributes to A.

[1] `https://archive.ics.uci.edu/dataset/203/yearpredictionmsd`

For the lower-level, we let $g_1(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - b\|^2$, which exhibits a L_{g_1}-Lipschitz continuous gradient, where $L_{g_1} = \lambda_{\max}(A^T A)$. Simultaneously, we set $g_2(\mathbf{x}) \equiv 0$. For the upper-level, we let $f_1(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$ and $f_2(\mathbf{x}) \equiv 0$. This configuration corresponds to finding the minimum norm solution. Now, our goal is to solve the following bilevel problem:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \tfrac{1}{2}\|\mathbf{x}\|^2 \\ \text{s.t.} \quad & \mathbf{x} \in \arg\min_{\mathbf{z}\in\mathbb{R}^n} \tfrac{1}{2}\|A\mathbf{z} - b\|^2 . \end{aligned} \tag{18}$$

We compare the performance of our Bisec-BiO with several existing methods to solve this problem, namely averaging iteratively regularized gradient method (a-IRG) (Kaushik and Yousefian, 2021), bilevel gradient SAM method (BiG-SAM) (Sabach and Shtern, 2017), minimal norm gradient method (MNG) (Beck and Sabach, 2014), and dynamic barrier gradient descent method (DBGD) (Gong and Liu, 2021). In this experiment, the feasible set of the lower-level problem is unbounded. Therefore, we cannot directly apply CG-BiO (Jiang et al., 2023). We use MATLAB function `lsqminnorm` to solve problem (18) and obtain the optimal values g^* and p^* for benchmarking purposes.

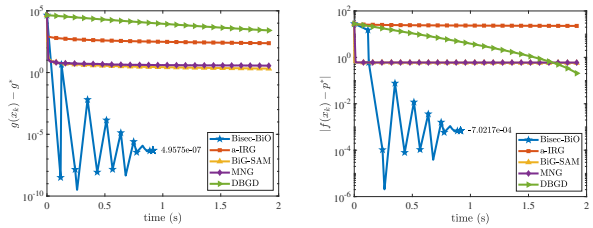

Figure 2: The performance of Bisec-BiO compared with other methods in MNP.

As Figure 2 demonstrates, our Bisec-BiO converges much faster than the other baseline methods for both the lower- and upper-level objectives, confirming our complexity results (see Table 1). Our method is shown in the figure as an oscillating curve, which depends on the feasibility of System (5) at each iteration point and the update mode of our method. Moreover, the left and right subfigures in Figure 2 show that the output of our algorithm meets the (ϵ_f, ϵ_g)-optimal solution criterion, as proven in Theorem 1.

5.2 Logistic Regression Problem (LRP)

We address the logistic regression binary classification problem using the 'a1a' dataset from LIBSVM[2], which contains $m = 30,956$ instances, each with $n = 123$ features. We randomly select a sample of $1,000$ instances and denote the feature matrix and labels as A and b, respectively.

For the lower-level, we let $g_1(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^m \log(1 + \exp(-a_i^\top \mathbf{x} b_i))$, which has a L_{g_1}-Lipschitz continuous gradient with $L_{g_1} = \frac{1}{4m}\lambda_{\max}(A^T A)$. Here, a_i represents an instance and $b_i \in \{-1, 1\}$ is the corresponding label. Additionally, we set $g_2(\mathbf{x}) = I_C(\mathbf{x})$, where $I_C(\mathbf{x})$ is the indicator function of $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \le \lambda\}$ with $\lambda = 10$. For the upper-level, we let $f_1(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$ and $f_2(\mathbf{x}) \equiv 0$, as well. We need to solve the following problem:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \tfrac{1}{2}\|\mathbf{x}\|^2 \\ \text{s.t.} \quad & \mathbf{x} \in \arg\min_{\mathbf{z}\in\mathbb{R}^n} \tfrac{1}{m}\sum_{i=1}^m \log(1 + \exp(-a_i^\top \mathbf{z} b_i)) \\ & + I_C(\mathbf{z}). \end{aligned} \tag{19}$$

In this experiment, we compare the performance of our Bisec-BiO with the methods proposed in Section 5.1 and the CG-based bilevel optimization method (CG-BiO) (Jiang et al., 2023). For benchmarking purposes, we utilize the Greedy FISTA algorithm (Liang et al., 2022) and MATLAB function `fmincon` to solve problem (18) and obtain the optimal values g^* and p^*, respectively. We employ the method proposed in Liu et al. (2020) to compute the proximal operator of $h_c := g_2 + \delta_c$ in problem (8). Their method is demonstrated to have a worst-case complexity of $O(n^2)$ and an observed complexity of $O(n)$ in practice.

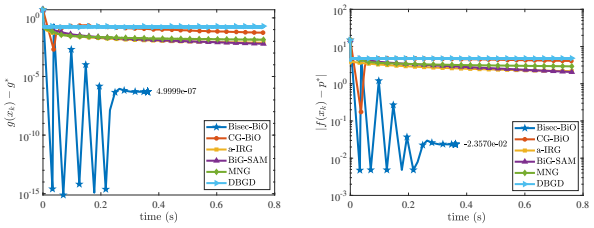

Figure 3: The performance of Bisec-BiO compared with other methods in LRP.

Figure 3 also shows that Bisec-BiO converges much faster than the other baseline methods for both the lower- and upper-level objectives. We have similar observations as in Figure 2.

6 CONCLUSION

In this paper, we address the problem of minimizing a composite convex function at the upper-

[2] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary/a1a.t`

level within the optimal solution set of a composite convex lower-level problem. We introduce a bisection algorithm designed to discover an ϵ_f-optimal solution for the upper-level objective and an ϵ_g-optimal solution for the lower-level objective. Our method attains a near-optimal convergence rate of $\tilde{\mathcal{O}}\left(\max\{\sqrt{L_{f_1}/\epsilon_f}, \sqrt{L_{g_1}/\epsilon_g}\}\right)$ for both upper- and lower-level objectives. Notably, this near-optimal rate aligns with the optimal rate observed in unconstrained smooth or composite optimization, neglecting the logarithmic term. We enhance convergence guarantees by imposing a Hölderian error bound assumption on the lower-level problem. Numerical experiments convincingly illustrate the substantial improvement our method offers over the state-of-the-art. In future research, we will explore the possibility of eliminating the logarithmic term from our complexity result.

Acknowledgements

Rujun Jiang is partly supported by the National Key R&D Program of China under grant 2023YFA1009300, National Natural Science Foundation of China under grants 12171100 and 72394364, and Natural Science Foundation of Shanghai 22ZR1405100. Jiulin Wang is supported by China Postdoctoral Science Foundation under grants BX20220085 and 2022M710798.

References

Samir Adly, Loïc Bourdin, and Fabien Caubet. On a decomposition formula for the proximal operator of the sum of two convex functions. *Journal of Convex Analysis*, 26(2):699–718, 2019.

Mostafa Amini and Farzad Yousefian. An iterative regularized incremental projected subgradient method for a class of bilevel optimization problems. In *2019 American Control Conference (ACC)*, pages 4069–4074. IEEE, 2019.

Heinz H. Bauschke, Minh N. Bui, and Xianfu Wang. Projecting onto the intersection of a cone and a sphere. *SIAM Journal on Optimization*, 28(3):2158–2188, 2018.

Amir Beck and Shoham Sabach. A first order method for finding minimal norm-like solutions of convex optimization problems. *Mathematical Programming*, 147(1-2):25–46, 2014.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2 (1):183–202, 2009.

Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

Nicholas Bishop, Long Tran-Thanh, and Enrico Gerding. Optimal learning from verified training data. *Advances in Neural Information Processing Systems*, 33:9520–9529, 2020.

Jérme Bolte, Trong Phong Nguyen, Juan Peypouquet, and Bruce W. Suter. From error bounds to the complexity of first-order descent methods for convex functions. *Mathematical Programming*, 165: 471–507, 2017.

J Frédéric Bonnans and Alexander Shapiro. *Perturbation analysis of optimization problems*. Springer Science & Business Media, 2013.

James V. Burke and Sien Deng. Weak sharp minima revisited, part ii: application to linear regularity and error bounds. *Mathematical Programming*, 104(2-3): 235–261, 2005.

Christine De Mol, Ernesto De Vito, and Lorenzo Rosasco. Elastic-net regularization in learning theory. *Journal of Complexity*, 25(2):201–230, 2009.

Stephan Dempe, Nguyen Dinh, Joydeep Dutta, and Tanushree Pandit. Simple bilevel programming and extensions. *Mathematical Programming*, 188:227–253, 2021.

Stephen Dempe, Nguyen Dinh, and Joydeep Dutta. Optimality conditions for a simple convex bilevel programming problem. *Variational Analysis and Generalized Differentiation in Optimization and Control: In Honor of Boris S. Mordukhovich*, pages 149–161, 2010.

Lior Doron and Shimrit Shtern. Methodology and first-order algorithms for solving nonsmooth and non-strongly convex bilevel optimization problems. *Mathematical Programming*, 201:521–558, 2023.

Dmitriy Drusvyatskiy and Adrian S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *Mathematics of operations research*, 43(3):919–948, 2018.

John Duchi. Elastic net projections. Technical report, Stanford University, 2009.

Joydeep Dutta and Tanushree Pandit. Algorithms for simple bilevel programming. *Bilevel Optimization: Advances and Next Challenges*, pages 253–291, 2020.

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*, pages 1568–1577. PMLR, 2018.

Michael P Friedlander and Paul Tseng. Exact regularization of convex programs. *SIAM Journal on Optimization*, 18(4):1326–1350, 2008.

Chengyue Gong and Xingchao Liu. Bi-objective trade-off with dynamic barrier gradient descent. *NeurIPS 2021*, 2021.

Pinghua Gong, Kun Gai, and Changshui Zhang. Efficient euclidean projections via piecewise root finding and its application in gradient projection. *Neurocomputing*, 74(17):2754–2766, 2011.

Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*, 2020.

Ruichen Jiang, Nazanin Abolfazli, Aryan Mokhtari, and Erfan Yazdandoost Hamedani. Conditional gradient-based method for bilevel optimization with convex lower-level problem. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.*, 2023.

Harshal D. Kaushik and Farzad Yousefian. A method with convergence rates for optimization problems with variational inequality constraints. *SIAM Journal on Optimization*, 31(3):2171–2198, 2021.

Puya Latafat, Andreas Themelis, Silvia Villa, and Panagiotis Patrinos. Adabim: An adaptive proximal gradient method for structured convex bilevel optimization. *arXiv preprint arXiv:2305.03559*, 2023.

Guoyin Li and Ting Kei Pong. Calculus of the exponent of kurdyka–łojasiewicz inequality and its applications to linear convergence of first-order methods. *Foundations of computational mathematics*, 18(5): 1199–1232, 2018.

Jingwei Liang, Tao Luo, and Carola-Bibiane Schonlieb. Improving "fast iterative shrinkage-thresholding algorithm": Faster, smarter, and greedier. *SIAM Journal on Scientific Computing*, 44(3):A1069–A1091, 2022.

Hongying Liu, Hao Wang, and Mengmeng Song. Projections onto the intersection of a one-norm ball or sphere and a two-norm ball or sphere. *Journal of Optimization Theory and Applications*, 187:520–534, 2020.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1), 2010.

Yura Malitsky. Chambolle-pock and tseng's methods: relationship and extension to the bilevel optimization. *arXiv preprint arXiv:1706.02602*, 2017.

Roey Merchav and Shoham Sabach. Convex bi-level optimization problems with non-smooth outer objective function. *arXiv preprint arXiv:2307.08245*, 2023.

Arkadij Semenovič Nemirovsky and David Borisovich Yudin. *Problem complexity and method efficiency in optimization*. Wiley, 1983.

Yurii Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Doklady Akademii Nauk*, 269(3):543–547, 1983.

Jong Shi Pang. Error bounds in mathematical programming. *Mathematical Programming*, 79(1-3): 299–332, 1997.

Nelly Pustelnik and Laurent Condat. Proximity operator of a sum of functions; application to depth map estimation. *IEEE Signal Processing Letters*, 24(12): 1827–1831, 2017.

Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. *Advances in neural information processing systems*, 32, 2019.

R. Tyrrell Rockafellar. *Convex Analysis*. Princeton Landmarks in Mathematics and Physics. Princeton University Press, 1970.

Emanuele Rodola, Andrea Torsello, Tatsuya Harada, Yasuo Kuniyoshi, and Daniel Cremers. Elastic net constraints for shape matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1169–1176, 2013.

Shoham Sabach and Shimrit Shtern. A first order method for solving convex bilevel optimization problems. *SIAM Journal on Optimization*, 27(2):640–660, 2017.

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*, pages 1723–1732. PMLR, 2019.

Yekini Shehu, Phan Tu Vuong, and Alain Zemkoho. An inertial extrapolation method for convex simple bilevel optimization. *Optimization Methods and Software*, 36(1):1–19, 2021.

Mikhail Solodov. An explicit descent method for bilevel convex optimization. *Journal of Convex Analysis*, 14(2):227–237, 2007.

Andreĭ Nikolaevich Tikhonov and V. I. A. K. Arsenin. *Solutions of ill-posed problems*. Wiley, 1977.

Jiali Wang, He Chen, Rujun Jiang, Xudong Li, and Zihao Li. Fast algorithms for stackelberg prediction game with least squares loss. In *International Conference on Machine Learning*, pages 10708–10716. PMLR, 2021.

Jiali Wang, Wen Huang, Rujun Jiang, Xudong Li, and Alex L Wang. Solving stackelberg prediction game with least squares loss via spherically constrained

least squares reformulation. In *International Conference on Machine Learning*, pages 22665–22679. PMLR, 2022.

Blake E Woodworth and Nati Srebro. Tight complexity bounds for optimizing composite objectives. *Advances in neural information processing systems*, 29, 2016.

Yao-Liang Yu. On decomposing the proximal map. *Advances in neural information processing systems*, 26, 2013.

Zirui Zhou and Anthony Man-Cho So. A unified approach to error bounds for structured convex optimization problems. *Mathematical Programming*, 165:689–728, 2017.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A THE FAST ITERATIVE SHRINKAGE THRESHOLDING ALGORITHM (FISTA) AND CONVERGENCE RESULTS

We adopt the Fast Iterative Shrinkage Thresholding Algorithm (FISTA) proposed in Beck and Teboulle (2009) as the APG method (see Definition 1) to solve problem (7). For any $L > 0$, define $p_L(\mathbf{y})$ as the unique minimizer of the following quadratic approximation of $\varphi(\mathbf{x})$ at a given point $\mathbf{y}$:

$$p_L(\mathbf{y}) := \arg\min_{\mathbf{x}\in\mathbb{R}^n} Q_L(\mathbf{x},\mathbf{y}) := \left\{\varphi_1(\mathbf{y}) + \langle \mathbf{x}-\mathbf{y}, \nabla\varphi_1(\mathbf{y})\rangle + \frac{L}{2}\|\mathbf{x}-\mathbf{y}\|^2 + \varphi_2(\mathbf{x})\right\},$$

where L actually plays the role of a step-size.

The classical FISTA scheme with a constant step-size for solving problem (7) is presented in Algorithm 2.

Algorithm 2 FISTA with constant step-size

Input: L_{φ_1}, $t_1 = 1$, $\mathbf{y}_1 = \mathbf{x}_0 \in \mathbb{R}^n$

1: **for** $k = 1, \cdots$ **do**
2: $\quad \mathbf{x}_k = p_{L_{\varphi_1}}(\mathbf{y}_k)$,
3: $\quad t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$,
4: $\quad \mathbf{y}_{k+1} = \mathbf{x}_k + \frac{t_k-1}{t_{k+1}}(\mathbf{x}_k - \mathbf{x}_{k-1})$,
5: $\quad k = k+1$.
6: **end for**

As discussed in Section 2, a potential limitation of this classical scheme is its dependence on the knowledge or computation of the Lipschitz constant L_{φ_1}, which may not be practical. To overcome this issue, Beck and Teboulle (2009) further proposed a variant of FISTA that incorporates a backtracking line search, we present it in Algorithm 3.

Algorithm 3 FISTA with backtracking line search

Input: $L_0 > 0$, $\eta > 1$, $t_1 = 1$, $\mathbf{y}_1 = \mathbf{x}_0 \in \mathbb{R}^n$

1: **for** $k = 1, \cdots$ **do**
2: $\quad$ Find the smallest nonnegative integer value i_k such that with $\bar{L} = \eta^{i_k} L_{k-1}$,

$$\varphi(p_{\bar{L}}(\mathbf{y}_k)) \le Q_{\bar{L}}(p_{\bar{L}}(\mathbf{y}_k), \mathbf{y}_k).$$

3: $\quad L_k = \eta^{i_k} L_{k-1}$,
4: $\quad \mathbf{x}_k = p_{L_k}(\mathbf{y}_k)$,
5: $\quad t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$,
6: $\quad \mathbf{y}_{k+1} = \mathbf{x}_k + \frac{t_k-1}{t_{k+1}}(\mathbf{x}_k - \mathbf{x}_{k-1})$,
7: $\quad k = k+1$.
8: **end for**

The next lemma shows the convergence results of the objective function under the FISTA scheme (Algorithm 2 or 3).

Lemma 2 ((Beck and Teboulle, 2009), Theorem 4.4.)**.** *Denote X_φ^* as the optimal solution set of problem (7) and $\mathbf{x}_\varphi^* \in X_\varphi^*$ be any optimal solution. Let $\mathbf{x}_0^\varphi \in \mathbb{R}^n$ be an initial point. Let $\{\mathbf{x}_k\}$ be the sequence generated by the FISTA scheme (Algorithm 2 or 3). Then for any $k \ge 1$, we have*

$$\varphi(\mathbf{x}_k) - \varphi(\mathbf{x}_\varphi^*) \le \frac{2\alpha L_{\varphi_1}}{(k+1)^2}\|\mathbf{x}_0^\varphi - \mathbf{x}_\varphi^*\|^2, \ \forall \mathbf{x}_\varphi^* \in X_\varphi^*.$$

Here, $\alpha = 1$ for the classical FISTA scheme (Algorithm 2), and $\alpha = \eta$ for the FISTA scheme with a backtracking line search (Algorithm 3), where η is the backtracking parameter. This lemma demonstrates that an ϵ-optimal solution of problem (7) can be obtained by Algorithm 2 or 3 within at most $\mathcal{O}(\sqrt{L_{\varphi_1}/\epsilon})$ iterations.

B MOTIVATING EXAMPLES

Many applications in machine learning and signal processing involve regularized problems, where the upper-level objectives represent the regularization terms, and the lower-level objectives consist of the loss functions and the additional constraint terms. We present some motivating examples below.

Example 2 (Minimum Norm Solution of Least Squares Regression Problem (MNP))**.** *Linear inverse problems aim to reconstruct a vector $\mathbf{x} \in \mathbb{R}^n$ from a set of measurements $b \in \mathbb{R}^m$ that satisfy the following relation: $b = A\mathbf{x} + \rho\varepsilon$, where $A : \mathbb{R}^n \to \mathbb{R}^m$ is a given linear mapping, $\varepsilon \in \mathbb{R}^m$ denotes an unknown noise vector, and $\rho > 0$ denotes its magnitude. Linear inverse problems can be solved using various optimization techniques, and we focus on the bilevel formulation (Beck and Sabach, 2014; Sabach and Shtern, 2017; Dempe et al., 2021; Latafat et al., 2023; Merchav and Sabach, 2023).*

The lower-level objective function in this formulation is given by

$$g(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - b\|^2 + I_C(\mathbf{x}),$$

where the set C is a closed convex set that can be chosen as $C = \mathbb{R}^n$, $C = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq 0\}$ or $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \leq \lambda\}$ for some $\lambda > 0$, and $I_C(\mathbf{x})$ is the indicator function of the set C, defined as $I_C(\mathbf{x}) = 0$ if $\mathbf{x} \in C$ and $I_C(\mathbf{x}) = +\infty$ if $\mathbf{x} \notin C$.

This problem may have multiple optimal solutions. Therefore, a natural choice is to consider the minimal norm solution problem, which seeks to find the optimal solution with the smallest Euclidean norm (Beck and Sabach, 2014; Sabach and Shtern, 2017; Latafat et al., 2023):

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2.$$

We then solve the bilevel optimization problem:

$$\begin{array}{ll} \min\limits_{\mathbf{x}\in\mathbb{R}^n} & \frac{1}{2}\|\mathbf{x}\|^2 \\ \text{s.t.} & \mathbf{x} \in \arg\min\limits_{\mathbf{z}\in\mathbb{R}^n} \frac{1}{2}\|A\mathbf{z} - b\|^2 + I_C(\mathbf{z}). \end{array}$$

For this example, the proximal mapping of h_c reduces to an orthogonal projection onto the ℓ_2-norm ball when $C = \mathbb{R}^n$. This scenario corresponds to the experiment in Section 5.1.

When $C = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq 0\}$, we have

$$\text{prox}_{h_c}(\mathbf{y}) = \frac{\sqrt{2c}}{\max\{\|P_C(\mathbf{y})\|, \sqrt{2c}\}} \cdot P_C(\mathbf{y}),$$

where $P_C(\mathbf{y}) = \max(\mathbf{y}, 0)$. This result is an implementation of Theorem 7.1 in Bauschke et al. (2018).

When $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \leq \lambda\}$, the proximal mapping of h_c simplifies to an orthogonal projection onto the intersection of a ℓ_2-norm ball and a ℓ_1-norm ball. This projection has a worst-case complexity of $O(n^2)$ and an observed complexity of $O(n)$ in practice (Liu et al., 2020). In this case, the lower-level objectives satisfy the Hölderian error bound assumption (Assumption 3) with $r = 2$.

Example 3 (Sparse Solution of Least Squares Regression Problem (SSP))**.** *Considering the same settings as in Example 2. To simplify the model and save computational resources and efficiency, we seek to reduce the number of features in the vector $\mathbf{x} \in \mathbb{R}^n$ that minimizes the linear inverse regression function $g(\cdot)$. This means that our goal is to find a sparse solution among all the minimizers of $g(\cdot)$. Therefore, any function that promotes sparsity can be used for this purpose. For example, the well-known elastic net regularization is a good choice (Zou and Hastie, 2005; Friedlander and Tseng, 2008; De Mol et al., 2009; Rodola et al., 2013; Amini and Yousefian, 2019; Merchav and Sabach, 2023). The elastic net regularization is defined as*

$$f(\mathbf{x}) = \|\mathbf{x}\|_1 + \frac{\alpha}{2}\|\mathbf{x}\|^2,$$

where $\alpha > 0$ regulates the trade-off between ℓ_1 and ℓ_2 norms.

For this example, the proximal mapping of h_c reduces to an orthogonal projection onto the elastic-net constraints when $C = \mathbb{R}^n$ (Duchi, 2009; Mairal et al., 2010; Gong et al., 2011; Rodola et al., 2013).

Example 4 (Logistic Regression Classification Problem (LRP))**.** *The goal of a binary classification problem is to establish a mapping from the feature vectors a_i to the target labels b_i. A common machine learning approach for this task is to minimize the logistic regression function of the given dataset (Amini and Yousefian, 2019; Gong and Liu, 2021; Jiang et al., 2023; Latafat et al., 2023; Merchav and Sabach, 2023). More precisely, we have a feature matrix $A \in \mathbb{R}^{m\times n}$ and a corresponding label vector $b \in \mathbb{R}^m$, where each $b_i \in -1, 1$. The logistic loss function is then given by*

$$g_1(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} \log(1 + \exp(-a_i^\top \mathbf{x} b_i)).$$

Over-fitting may occur when the number of features is not negligible relative to the number of instances m. A common way to address this problem is to regularize the logistic objective function with a specific function or add a constraint (Jiang et al., 2023; Merchav and Sabach, 2023). For example, we can choose $g_2(\mathbf{x}) = I_C(\mathbf{x})$, where $I_C(\mathbf{x})$ is the indicator of the set $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \le \lambda\}$ as shown in Example 2.

Similarly, this problem may have multiple optimal solutions. Therefore, it is natural to consider the minimal norm solution problem with the smallest Euclidean norm as described in Example 2. Now, we need to solve the following bilevel optimization problem (Gong and Liu, 2021; Jiang et al., 2023; Latafat et al., 2023):

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \tfrac{1}{2}\|\mathbf{x}\|^2 \\ \text{s.t.} \quad & \mathbf{x} \in \arg\min_{\mathbf{z}\in\mathbb{R}^n} \tfrac{1}{m}\sum_{i=1}^{m} \log(1 + \exp(-a_i^\top \mathbf{z} b_i)) + I_C(\mathbf{z}). \end{aligned}$$

The proximal mappings of h_c are the same for the choices of the set C in Example 2. In particular, when $C = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_1 \le \lambda\}$, this scenario corresponds to the experiment in Section 5.2.

Example 5 (Sparse Solution of Logistic Regression Classification Problem (SSLRP))**.** *Considering the same settings as in Example 4, and we seek to reduce the features in the vector $\mathbf{x} \in \mathbb{R}^n$ that minimizes the logistic regression function with a regularization term. For this purpose, we can choose the elastic-net regularization, which is proposed in Example 3.*

Similarly, for this example, the proximal mapping of h_c is an orthogonal projection onto the elastic-net constraints when $C = \mathbb{R}^n$, as shown in Example 3.

C PROOF OF THE MAIN THEOREMS

C.1 Proof of Theorem 1

Proof. We first show that $\hat{\mathbf{x}}$ is an (ϵ_f, ϵ_g)-optimal solution of problem (1). In Step 10, we let $c = u$. Then Condition (11) is not satisfied. According to Lemma 1, the inequality $g(\hat{\mathbf{x}}) \le g^* + \epsilon_g$ holds. Next, we prove $f(\hat{\mathbf{x}}) \le p^* + \epsilon_f$ also holds. We break the proof into two cases.

Case (1), if $u \le p^*$, then we have $f(\hat{\mathbf{x}}) = u \le p^*$.

Case (2), if $u > p^*$, then p^* lies on $[l, u]$ since $l \le p^*$ always holds. Therefore, we have

$$f(\hat{\mathbf{x}}) = u \le u + p^* - l \le p^* + \epsilon_f,$$

where the last inequality is from the stop criterion that $u - l \le \epsilon_f$. To sum up, the point $\hat{\mathbf{x}}$ is an (ϵ_f, ϵ_g)-optimal solution of problem (1). In the following, we present the total operation complexity of our method.

In Step 1, we invoke APG oracles to obtain initial bounds l and u. To obtain the initial lower bound of p^*, we invoke the APG oracle $\tilde{\mathbf{x}}_f$=APG$(f_1, f_2, L_{f_1}, \mathbf{x}_0^f, \epsilon_f/2)$ to solve problem (9). By Lemma 2, this can be done within $\mathcal{O}\left(\sqrt{L_{f_1}/\epsilon_f}\right)$ iterations. The corresponding initial lower bound is $l = f(\tilde{\mathbf{x}}_f) - \epsilon_f/2$. To obtain the initial upper bound of $p^*_{\epsilon_g}$, we invoke the APG oracle $\tilde{\mathbf{x}}_g$=APG$(g_1, g_2, L_{g_1}, \mathbf{x}_0^g, \epsilon_g/2)$ to solve problem (2). Similarly, we

can approximately solve problem (2) within $\mathcal{O}\left(\sqrt{L_{g_1}/\epsilon_g}\right)$ iterations. The corresponding initial upper bound is given by $u = f(\tilde{\mathbf{x}}_g)$.

In Step 3, we invoke APG oracle $\tilde{\mathbf{x}}_c$=APG$(g_1, h_c, L_{g_1}, \mathbf{x}_0^c, \epsilon_g/2)$ to solve problem (6). According to Lemma 2, this can be done within $\mathcal{O}\left(\sqrt{L_{g_1}/\epsilon_g}\right)$ iterations. The number of invoking APG oracle does not exceed

$$\left\lceil \log_2 \frac{u-l}{\epsilon_f} \right\rceil_+ = \left\lceil \log_2 \frac{f(\tilde{\mathbf{x}}_g) - f(\tilde{\mathbf{x}}_f) + \epsilon_f/2}{\epsilon_f} \right\rceil_+ = \mathcal{O}\left(\log \frac{1}{\epsilon_f}\right)$$

where l and u are initial lower and upper bounds, and $\lceil a \rceil_+$ represents the smallest non-negative integer that is no less than a.

Thus, the total number of evaluations of the function values f_1, f_2, g_1, and g_2, the gradients ∇f_1 and ∇g_1, and the calls of the proximal mapping concerning function h_c does not exceed T, where

$$T = \mathcal{O}\left(\sqrt{\frac{L_{f_1}}{\epsilon_f}}\right) + \mathcal{O}\left(\sqrt{\frac{L_{g_1}}{\epsilon_g}}\right) + \mathcal{O}\left(\sqrt{\frac{L_{g_1}}{\epsilon_g}}\right) \cdot \mathcal{O}\left(\log \frac{1}{\epsilon_f}\right) = \tilde{\mathcal{O}}\left(\max\left\{\sqrt{\frac{L_{f_1}}{\epsilon_f}}, \sqrt{\frac{L_{g_1}}{\epsilon_g}}\right\}\right),$$

where $\tilde{\mathcal{O}}$ suppresses a logarithmic term. □

C.2 Proof of Proposition 1

Proof. By Assumption 2(i), the set X_g^* is closed and compact. Then we can let $\hat{\mathbf{x}}^* = \arg\min_{\mathbf{x} \in X_g^*} \|\mathbf{x} - \hat{\mathbf{x}}\|$ such that $\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^*\| = \operatorname{dist}(\hat{\mathbf{x}}, X_g^*)$. It can be easily demonstrated that X_g^* is a convex set, ensuring the well-definedness of $\hat{\mathbf{x}}^*$. By Assumption 3, we have

$$\frac{\alpha}{r}\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^*\|^r \le g(\hat{\mathbf{x}}) - g^* \le \epsilon_g \implies \|\hat{\mathbf{x}} - \hat{\mathbf{x}}^*\| \le \left(\frac{r\epsilon_g}{\alpha}\right)^{\frac{1}{r}}. \tag{20}$$

By Assumption 2, it follows that f is B_f-Lipschitz continuous on $\operatorname{dom}(g_2)$ (see (15)). Combining this result with (20), we have

$$f(\hat{\mathbf{x}}) - p^* \ge f(\hat{\mathbf{x}}) - f(\hat{\mathbf{x}}^*) \ge -B_f\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^*\| \ge -B_f\left(\frac{r\epsilon_g}{\alpha}\right)^{\frac{1}{r}},$$

where the first inequality is from that p^* is the optimal value of problem (1) and $\hat{\mathbf{x}}^* \in X_g^*$. □

C.3 Proof of Corollary 1

Proof. Let $\hat{\mathbf{x}}$ be the output of Algorithm 1. Then $\hat{\mathbf{x}}$ is an (ϵ_f, ϵ_g)-optimal solution of (1) satisfying

$$f(\hat{\mathbf{x}}) - p^* \le \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \le \epsilon_g.$$

By Proposition 1 and the setting $\epsilon_g = \frac{\alpha}{\gamma}\left(\frac{\epsilon_f}{B_f}\right)^r$, we have

$$f(\hat{\mathbf{x}}) - p^* \ge -B_f\left(\frac{r\epsilon_g}{\alpha}\right)^{\frac{1}{r}} = -\epsilon_f.$$

Then according to Theorem 1, with an operation complexity $\tilde{\mathcal{O}}\left(1/\sqrt{\epsilon_f^r}\right)$, we can find an $\hat{\mathbf{x}}$ such that

$$|f(\hat{\mathbf{x}}) - p^*| \le \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \le \epsilon_g.$$

□

C.4 Proof of Proposition 2

Proof. Since $\hat{\mathbf{x}}$ satisfies $g(\hat{\mathbf{x}}) - g^* \le \epsilon_g$, then $\hat{\mathbf{x}}$ is a feasible solution of (4) with $\epsilon = \epsilon_g$, and $f(\hat{\mathbf{x}})$ is an upper bound of $p_{\epsilon_g}^*$ (see Lemma 1). Applying Assumption 4 with $\epsilon = \epsilon_g$, we have

$$f(\hat{\mathbf{x}}) - p^* \ge p_{\epsilon_g}^* - p^* = v(\epsilon_g) - v(0) \ge -L_{\epsilon_g}\epsilon_g.$$

□

C.5 Proof of Corollary 2

Proof. This proof closely resembles the one presented in Corollary 1. Nevertheless, for the sake of thoroughness, we provide a complete exposition. Let $\hat{\mathbf{x}}$ be the output of Algorithm 1. Then $\hat{\mathbf{x}}$ is an (ϵ_f, ϵ_g)-optimal solution of (1) satisfying

$$f(\hat{\mathbf{x}}) - p^* \leq \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g.$$

By Proposition 2 and the setting $\epsilon_f = L_{\epsilon_g}\epsilon_g$, we have

$$f(\hat{\mathbf{x}}) - p^* \geq -L_{\epsilon_g}\epsilon_g = -\epsilon_f.$$

Then according to Theorem 1, with an operation complexity $\tilde{\mathcal{O}}\left(\max\{1/\sqrt{L_{\epsilon_g}\epsilon_g}, 1/\sqrt{\epsilon_g}\}\right)$, we can find an $\hat{\mathbf{x}}$ such that

$$|f(\hat{\mathbf{x}}) - p^*| \leq \epsilon_f,\ g(\hat{\mathbf{x}}) - g^* \leq \epsilon_g.$$

□

D ADDITIONAL NUMERICAL EXPERIMENTS

We consider the SSP problem from Example 3 on the YearPredictionMSD dataset, setting $C = \mathbb{R}^n$ and $\alpha = 0.02$. Our objective is to solve the following bilevel problem:

$$\begin{array}{ll} \min\limits_{\mathbf{x}\in\mathbb{R}^n} & \frac{\alpha}{2}\|\mathbf{x}\|^2 + \|\mathbf{x}\|_1 \\ \text{s.t.} & \mathbf{x} \in \arg\min\limits_{\mathbf{z}\in\mathbb{R}^n} \frac{1}{2}\|A\mathbf{z} - b\|^2. \end{array} \tag{21}$$

We compare the performance of our Bisec-BiO method with the averaging iteratively regularized gradient (a-IRG) method (Kaushik and Yousefian, 2021). It's worth noting that a-IRG can handle non-smooth upper-level objectives, a capability that other methods lack. For benchmarking purposes, we use the MATLAB functions `lsqminnorm` and `fmincon` to obtain the optimal values g^* and p^*, respectively. Moreover, we adopt the method proposed by Gong et al. (2011) to compute the proximal operator of $h_c := g_2 + \delta_c$ in problem (8). The other settings are consistent with Section 5.1.

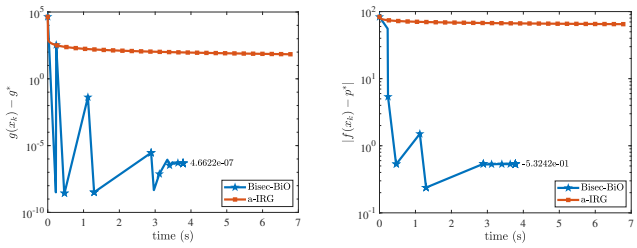

Figure 4: The performance of Bisec-BiO compared with other methods in SSP.

As illustrated in Figure 4, Bisec-BiO converges significantly faster than a-IRG for both lower- and upper-level objectives. The numbers on the right-hand side of the last iterates of our method (denoted as $\hat{\mathbf{x}}$) represent the differences between $g(\hat{\mathbf{x}}) - g^*$ and $f(\hat{\mathbf{x}}) - p^*$, respectively. This confirms that $\hat{\mathbf{x}}$ is an (ϵ_f, ϵ_g)-optimal solution.