Accelerated Fully First-Order Methods for Bilevel and Minimax Optimization

Chris Junchi Li$^\diamond$

Department of Electrical Engineering and Computer Sciences$^\diamond$
University of California, Berkeley

June 17, 2024

Abstract

We present in this paper novel accelerated fully first-order methods in *Bilevel Optimization* (BiO). Firstly, for BiO under the assumption that the lower-level functions admit the typical strong convexity assumption, the *(Perturbed) Restarted Accelerated Fully First-order methods for Bilevel Approximation* ((P)RAF^{2}BA) algorithm leveraging *fully* first-order oracles is proposed, whereas the algorithm for finding approximate first-order and second-order stationary points with state-of-the-art oracle query complexities in solving complex optimization tasks. Secondly, applying as a special case of BiO the *nonconvex-strongly-convex* (NCSC) minimax optimization, PRAF^{2}BA rediscovers *perturbed restarted accelerated gradient descent ascent* (PRAGDA) that achieves the state-of-the-art complexity for finding approximate second-order stationary points. Additionally, we investigate the challenge of finding stationary points of the hyper-objective function in BiO when lower-level functions lack the typical strong convexity assumption, where we identify several regularity conditions of the lower-level problems that ensure tractability and present hardness results indicating the intractability of BiO for general convex lower-level functions. Under these regularity conditions we propose the *Inexact Gradient-Free Method* (IGFM), utilizing the *Switching Gradient Method* (SGM) as an efficient sub-routine to find an approximate stationary point of the hyper-objective in polynomial time. Empirical studies for real-world problems are provided to further validate the outperformance of our proposed algorithms.

1 Introduction

Bilevel optimization (BiO) has received increasing attention owing to its remarkable capability in addressing crucial machine learning tasks by revealing the inner structure of many (otherwise oblique) machine learning optimization problems, such as meta-learning [FFS$^+$18, BHTV19, JLLP20, RL17, HAMS21], hyperparameter optimization [FFS$^+$18, Ped16, FH19, SCHB19, GFPS20, AM22a], continual learning [PLSS21], out-of-distribution learning [ZLP$^+$22], adversarial training [GPAM$^+$20, SND18, WCJ$^+$21, LJJ20a, LJJ20b, WL20], composite optimization [GHZY21], reinforcement learning [KT99, HWWY23, KZH$^+$21, SZB20], causal learning [JV22, LSR$^+$22, ABGLP19], neural architecture search [LSY19, WGS$^+$22, ZL17, ZSP$^+$21], etc. Formally, BiO aims to optimize the *upper-level* (UL) function $f(x,y)$ under the constraint that y is minimized with respect to the *lower-level* (LL) function $g(x,y)$ on a closed convex set $\mathcal{Y} \subseteq \mathbb{R}^{d_y}$. Mathematically, it can be formulated as

$$\min_{x\in\mathbb{R}^{d_x}, y\in Y^*(x)} f(x,y) \qquad \text{where} \qquad Y^*(x) \triangleq \arg\min_{y\in\mathcal{Y}} g(x,y) \text{ is the } \textit{LL solution mapping} \tag{1}$$

Let LL value function be $g^*(x) \triangleq \min_{y\in\mathcal{Y}} g(x,y)$. Problem (1) can be transformed via hyper-objective approaches [Dem02, DZ20, LMY$^+$20, LLZZ21]

$$\min_{x\in\mathbb{R}^{d_x}} \left\{ \varphi(x) \triangleq \min_{y\in Y^*(x)} f(x,y) \right\} \tag{2}$$

where $\varphi(x)$ is called the *hyper-objective function* of BiO problem (1). It transforms the problem into the composition of a simple BiO [SS17] w.r.t. the LL variable y and an unconstrained single-level optimization w.r.t. the UL variable x. This reformulation naturally leads to two foundational questions. The first question involves

P1: Find an optimal LL variable $\widehat{y} \in Y^*(\widehat{x})$ such that $\varphi(\widehat{x}) = f(\widehat{x}, \widehat{y})$ for a given $\widehat{x}$

The second question involves

P2: Find a UL variable $\widehat{x}$ that is a stationary point of $\varphi(x)$

BiO with LLSC. When the LL function is strongly convex, both questions previously proposed are relatively easy to solve. The lower-level strong convexity (LLSC) ensures $Y^*(x)$ to be a singleton, and therefore simplifies (2) into $\varphi(x) = f(x, y^*(x))$, where the LL optimal solution $y^*(x) = \arg\min_{y\in\mathcal{Y}} g(x, y)$ can be found via gradient descent on g. For simplicity we assume in the LLSC case $\mathcal{Y} = \mathbb{R}^{d_y}$. In this case, (1) is translated into

$$\begin{aligned} \min_{x\in\mathbb{R}^{d_x}} \quad & \varphi(x) \triangleq f(x, y^*(x)) \\ \text{s.t.} \quad & y^*(x) = \arg\min_{y\in\mathbb{R}^{d_y}} \; g(x, y) \end{aligned} \tag{3}$$

where the UL function $f(x, y)$ is smooth and possibly nonconvex, and the LL function $g(x, y)$ is smooth and (strongly) convex with respect to y for any given x. In this case, the implicit function theorem indicates

$$\nabla\varphi(x) = \nabla_x f(x, y^*(x)) - \nabla^2_{xy} g(x, y^*(x)) \left[\nabla^2_{yy} g(x, y^*(x))\right]^{-1} \nabla_y f(x, y^*(x)) \tag{4}$$

Then one can apply the gradient step with $\nabla\varphi(x)$ to find a UL stationary point. This forms the basis of the classical hyper-objective approaches for BiO with LLSC [JYL21].

Our goal is to establish the theoretical convergence guarantee to this problem, with access to *fully first-order oracles* of $f(x, y)$ and $g(x, y)$ in the sense that there is *no access* to second-order information such as Jacobian- or Hessian-vector-product oracle, where we will be assuming $g(\cdot, y)$ is μ-strongly convex for some $\mu > 0$ which is shared across all $y \in \mathbb{R}^{d_y}$. Additional smoothness conditions posed on f and g capture the smoothness of the overall hyper-objective function $\varphi(x)$.

Minimax Optimization. An important special case of the BiO problem (3)—the problem of minimax optimization, where $g = -f$ in the LL problem (3)—has been extensively studied in the literature [LLC22, LJJ20a]. Seemingly the first in literature, we are able to show that `PRAF`^{2}`BA` rediscovers *perturbed restarted accelerated gradient descent ascent* (`PRAGDA`) recently proposed by [YLL+23] that achieves the state-of-the-art complexity for finding approximate second-order stationary points in *nonconvex-strongly-concave* (NCSC) minimax optimization, where the parameter flexibility of the algorithm enhances its adaptability to diverse problem settings.

BiO without LLSC. In many machine learning applications, however, the LLSC condition our accelerated methods heavily relied upon may not hold, and it is hence interesting to further investigate in BiO without LLSC, but only LL convexity. We formulate the LL optimality and UL stationarity as valid criteria for BiO without LLSC, which are necessary for an optimistic optimal solution [DKK06]. Further, we prove that when the LL function satisfies either the gradient dominance condition or the weak sharp minimum condition, the hyper-objective $\varphi(x)$ is Lipschitz and thus Clarke differentiable [§4.1]. We provide hardness results to show that BiO without LLSC is generally intractable. Our analysis highlights the importance of sharpness in LL functions [§4.2]. We propose novel polynomial time algorithms for BiO with LL convexity under either the *gradient dominance* or the *weak sharp minimum* condition [§4.3].

1.1 Contributions

This paper provides a comprehensive study of BiO with and without the LLSC assumption.

(i) For BiO with LLSC, we illustrate that our acceleration framework can be effectively incorporated into the idea of fully first-order methods, improving the dependency on ϵ from ϵ^{-2} to $\epsilon^{-1.75}$. The *(Perturbed) Restarted Accelerated Fully First-order methods for Bilevel Approximation* ((P)RAF^{2}BA) introduced in this paper aims at solving (nonconvex-strongly-convex) BiO problems with effectiveness and efficiency. By leveraging fully first-order oracles and incorporating acceleration techniques, (P)RAF^{2}BA algorithm finds approximate first-order stationary points and second-order stationary points of the hyper-objective function at improved oracle query complexities [§2].

(ii) For NCSC minimax optimization, PRAF^{2}BA rediscovers PRAGDA that achieves the state-of-the-art complexity for finding approximate second-order stationary points, where the parameter flexibility of the algorithm enhances its adaptability to diverse problem settings [§3].

(iii) For BiO without LLSC, we compare the tractability and intractability results under different assumptions on the LL function. In particular, we introduce several key regularity conditions that can confer tractability, without which we provide hardness results to show the intractability of this problem. Novel algorithms with non-asymptotic convergence are also proposed [§4].

(iv) Using real-world datasets, we conduct empirical results including tasks of hyperparameter optimization, data hypercleaning and adversarial training, support our theoretical results and showcasing the superiority of our methods [§A].

1.2 Related Works

For BiO with LLSC, representative methods include *approximate implicit differentiation* (AID) [Dom12, GW18, Ped16, FFS+18, GFPS20, JYL21] and *iterative differentiation* (ITD) [GFC+16, FDFP17, SCHB19, BLPSF21] that have non-asymptotically convergence to a UL stationary point. In particular, Ghadimi and Wang [GW18] introduced a convergence rate for the AID approach under convex $f(x, y)$, analyzing a gradient descent-based accelerated algorithm. Due to their popularity, many improvements to AID and ITD have also been proposed [CSXY22, HWWY23, KZH+21, YJL21, JL23, JYL21, JLLY22, DAVM22]. Among them, Ji et al. [JYL21, JLLY22] improved upon this with their iterative differentiation (ITD) method, refining complexity analysis and providing insights into a randomized version. Hong et al. [HWWY23] proposed the TTSA algorithm, offering a single-loop solution for alternating variable updates, notably applicable to randomized reinforcement learning scenarios. For stochastic bilevel problems, various methods like BSA [GW18], TTSA [HWWY23], SUSTAIN [KZH+21], stocBiO [JYL21], and ALSET [CSY21] have been proposed, pushing the frontier with variance reduction and momentum techniques [JL23, LHH22, KKWN23]. While much focus has been on first-order stationary points in BiO, the pursuit of second-order stationary points remains largely underexplored. Until recently, Huang et al. [HJML22] introduced a perturbed algorithm to find approximate second-order stationary points, combining gradient descent with conjugate gradient methods. The algorithm utilizes gradient descent to approximate the solution of the LL minimization problem and employs conjugate gradient to solve for Hessian-vector products, with gradient descent applied in the outer loop. Indeed for classical optimization problems, second-order methods such as those proposed in [NP06, CRS17] have been employed to achieve ϵ-accurate second-order stationary points (SOSPs) in single-level optimization with a complexity of $\mathcal{O}(\epsilon^{-1.5})$—however, computationally expensive operations such as estimating the inverse of Hessian matrices are involved—and recent literature has focused on first-order methods to obtain an approximate $\big(\epsilon, O(\kappa^{2.5}\sqrt{\epsilon})\big)$-SOSP where κ denotes the condition number specified in §2, achieving a best-known complexity of $\tilde{\mathcal{O}}(\epsilon^{-1.75})$ in terms of gradient and Hessian-vector products [AAZB$^+$17, CDHS18, CDHS17, JGN$^+$17, JNJ18, LL23]. For more recent progress on BiO under nonsmooth LL function, we refer the readers to [LM23, LM24]. For more on second-order analysis for bilevel optimization, we refer to [SYL$^+$23, DSAP22].

For BiO problem in the absence of LLSC, [AM22b] showed that one can extend AID by replacing the inverse in (4) with the Moore-Penrose inverse under the Morse-Bott condition on the manifold $\{y \in \mathbb{R}^{d_y} : \nabla_y f(x,y) = 0\}$. [LLZZ21, LMY$^+$20] extended ITD by proposing various methods to update the LL variable. However, all the methods mentioned above are limited to asymptotic convergence to an LL optimal solution and lack analysis for finding a UL stationary point. Due to the challenge of directly optimizing the hyper-objective, some concurrent works [LYW$^+$22, SJGL22] reformulate Problem (1) via the value-function approach and show non-asymptotic convergence to the KKT points of this equivalent problem. However, since classical constraint qualifications provably fail for the reformulated problem [YZ95], the KKT condition is not even a necessary condition for a local minimum.[1] In contrast, a UL stationary point is always a necessary condition. More related works on this thread include [LLY$^+$21, SC23, XLC23].

Minimax optimization as an important special case of bilevel optimization is pivotal in machine learning applications like GAN training [GPAM$^+$20, ACB17], adversarial learning [GSS14, SND18], and optimal transport [LFH$^+$20, HML21], has garnered attention. Nouiehed et al. [NSH$^+$19], Jin et al. [JNJ20] explored the complexity of Multistep Gradient Descent Ascent (GDmax), while Lin et al. [LJJ20a], Lu et al. [LTHC20] provided the first convergence analysis for the gradient descent ascent (GDA) algorithm. Luo et al. [LYHZ20] extended stochastic variance reduction techniques, achieving optimal complexity bounds in specific cases. Recent work by Luo et al. [LLC22] and Chen et al. [CHL$^+$23] introduced cubic-regularized Newton methods for local minimax point convergence. Despite these strides, non-asymptotic convergence rates for local minimax points remain relatively unexplored, presenting a compelling area for future work.

Notation. For A being a real symmetric matrix, let $\lambda_{\max}(A)$ (resp. $\lambda_{\min}(A)$) denote its largest (resp. smallest) eigenvalue, and also $\kappa(A) = \lambda_{\max}(A)/\lambda_{\min}(A)$ denote its condition number. For real asymmetric matrix A', let $\sigma_{\max}(A')$ to be the largest singular value and $\sigma_{\min}^{+}(A')$ the smallest nonzero singular value. Let $\|\cdot\|$ denote either the spectral norm of matrices, or the Euclidean ℓ_2-norm of a vector, and $z_{[j]}$ denote the j-th coordinate of vector z. Denote $\mathbb{B}_\delta(z) = \{z' : \|z' - z\| \le \delta\}$ the closed Euclidean ball centered at z with radius δ, and $\mathbb{B}_\delta \triangleq \mathbb{B}_\delta(0)$ the ball centered at the origin. Denote $Gc(f,\epsilon)$, $JV(f,\epsilon)$ and $HV(f,\epsilon)$ as the oracle complexities of gradients, Jacobian-vector products and Hessian-vector products corresponding to function f, respectively. For two positive

[1] See, e.g., Example D.1 in §D.2.

sequences $\{a_n\}$ and $\{b_n\}$ we denote $a_n = \Omega(b_n)$ (resp. $a_n = \mathcal{O}(b_n)$) if $a_n \geq Cb_n$ (resp. $a_n \leq Cb_n$) for all n, and also $a_n = \Theta(b_n)$ if both $a_n = \Omega(b_n)$ and $a_n = \mathcal{O}(b_n)$ hold for some absolute constant $C > 0$, and $\widetilde{\mathcal{O}}(\cdot)$ or $\widetilde{\Omega}(\cdot)$ is adopted in turn when C incorporates a polylogarithmic factor in problem parameters.

2 Accelerated Fully First-Order Bilevel Optimization with LLSC

In this section, we present a new algorithm member for accelerating first-order methods for BiO, namely the *(Perturbed) Restarted Accelerated Fully First-order methods for Bilevel Approximation*, abbreviated as (P)RAF^{2}BA. Recent work [KKWN23] considers the first-order approximation for BiO problem (3) where they introduce the auxiliary function as follows

$$\mathcal{L}_\lambda(x,y) \triangleq f(x,y) + \lambda \left(g(x,y) - \min_{z \in \mathbb{R}^{d_y}} g(x,z) \right) \tag{5}$$

where $\lambda > 0$ is the regularization parameter. Under proper smoothness condition, taking $\lambda \geq 2\kappa$ where $\kappa \triangleq \ell/\mu$ to be specified later in Assumption 1 leads to $\mathcal{L}_\lambda(x,y)$ being strongly convex in y for any given x, which implies that the function $\mathcal{L}_\lambda^*(x) \triangleq \min_{y\in\mathbb{R}^{d_y}} \mathcal{L}_\lambda(x,y)$ is smooth [Dan12]. By setting λ appropriately—often growing with inverse precision ϵ—the approximate *first-order* (FOSP) and *second-order stationary points* (SOSP) of the objective $\varphi(x) \triangleq f(x, y^*(x))$ in the BiO problem (3) are sufficiently close to the corresponding stationary points. This implies that we can address the BiO problem by considering the minimization problem

$$\min_{x\in\mathbb{R}^{d_x}} \left\{ \mathcal{L}_\lambda^*(x) \triangleq \min_{y\in\mathbb{R}^{d_y}} \mathcal{L}_\lambda(x,y) \right\} \tag{6}$$

The expression of $\mathcal{L}_\lambda(x,y)$ in (5) suggests we can solve problem (6) by only accessing the first-order oracles of $f(x,y)$ and $g(x,y)$. Based on this idea, [KKWN23] proposed, among many other methods, a nonstochastic fully first-order method for finding ϵ-first-order stationary points of $\varphi(x)$ with a first-order oracle complexity of ϵ^{-3}.

- **Relationship with [YLL$^+$23].** Closely related to this part is [YLL$^+$23] by same (extended) group of authors [YLL$^+$23], which successfully accelerates an alternative family of algorithm—the inexact hypergradient method—for solving BiO problem with LLSC. We will carefully point out the connections between the two families, especially on their equivalence in the minimax optimization setting.

- **Key Ingredients due to [CMZ23].** Very recently and concurrent to our work [YLL$^+$23], Chen et al. [CMZ23] revisits the fully first-order methods of [KKWN23] and improves the first-order oracle complexity upper bound to $\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$.[2] The key observation is that the Lipschitz constant of the gradient of $\mathcal{L}_\lambda^*(x)$—defined as in (6)—can be set to be *not* dependent on λ, as the parameter grows. By further assuming that $g(x,y)$ admits Lipschitz continuous third-order derivatives, [CMZ23] also provided a perturbed first-order method to find $(\epsilon, \mathcal{O}(\kappa^{2.5}\sqrt{\epsilon}))$-second-order stationary points of $\Phi(x)$ within $\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$ first-order oracle complexity. *We illustrate that our acceleration framework can be effectively incorporated into the idea of fully first-order methods, improving the dependency on ϵ from ϵ^{-2} to $\epsilon^{-1.75}$.*

[2] We became aware of the work [CMZ23] around two to three months after the initial arXiv posting of [YLL$^+$23].

In §2.1 we conduct the technical overview and establish the basic notions, assumptions and algorithmic subroutines for the problem setting. §2.2 presents the `(P)RAF`^{2}`BA` algorithm with theoretical complexity bounds for accelerated fully first-order methods, highlighting the improvements in convergence rates and algorithmic frameworks compared to existing approaches. We delegate a complete proofs of theorems with detailed analysis presented in this section to §2.3, §2.4, §B.

2.1 Technical Preliminaries

In this subsection, we proceed to present the notation and assumptions necessary for our problem setting. Immediately afterward, we establish convergence of the algorithmic subroutine of *accelerated gradient descent* (`AGD`).

We first revisit the formal definition of an ϵ-first-order stationary point as well as an (ϵ, τ)-second-order stationary point of a twice differentiable function $\varphi(x)$ for any prescribed $\epsilon, \tau > 0$, as follows:

Definition 1 (Approximate first-order stationary point). *Call x an* ϵ-first-order stationary point *of $\varphi(x)$ if $\|\nabla\varphi(x)\|_2 \le \epsilon$.*

Definition 2 (Approximate second-order stationary point). *Call x an* (ϵ, τ)-second-order stationary point *of $\varphi(x)$ if $\|\nabla\varphi(x)\|_2 \le \epsilon$ and $\lambda_{\min}(\nabla^2\varphi(x)) \ge -\tau$.*

We turn to introduce some basic lemmas as follows. First and foremost, we introduce the following list of assumptions, which is core for our theoretical guarantees to hold:

Assumption 1. *The UL function $f(x, y)$ and LL function $g(x, y)$ satisfy the following conditions:*

(i) Function $g(x, y)$ is three times differentiable and μ-strongly convex with respect to y for any fixed x

(ii) Function $f(x, y)$ is twice differentiable and M-Lipschitz continuous with respect to y

(iii) Gradient $\nabla f(x, y)$ and $\nabla g(x, y)$ are ℓ-Lipschitz continuous with respect to x and y

(iv) The Jacobians $\nabla^2_{xy} f(x, y)$, $\nabla^2_{xy} g(x, y)$ and Hessians $\nabla^2_{xx} f(x, y)$, $\nabla^2_{yy} f(x, y)$, $\nabla^2_{yy} g(x, y)$ are ρ-Lipschitz continuous with respect to x and y

(v) The third-order derivatives $\nabla^3_{xyx} g(x, y), \nabla^3_{yxy} g(x, y)$ and $\nabla^3_{yyy} g(x, y)$ are ν-Lipschitz continuous with respect to x and y

We then show that $\varphi(x)$ admits Lipschitz continuous gradients and Lipschitz continuous Hessians, as established in the next lemma:

Lemma 1. *Suppose Assumption 1 holds, then*

(i) $\varphi(x)$ is $\widetilde{L}$-gradient Lipschitz continuous; that is, $\|\nabla\varphi(x) - \nabla\varphi(x')\| \le \widetilde{L}\|x - x'\|$ for any $x, x' \in \mathbb{R}^{d_x}$ where $\widetilde{L} = \mathcal{O}(\kappa^3)$

$$\widetilde{L} \triangleq \ell + \frac{2\ell^2 + \rho M}{\mu} + \frac{\ell^3 + 2\rho\ell M}{\mu^2} + \frac{\rho\ell^2 M}{\mu^3}$$

(ii) $\varphi(x)$ *is* $\widetilde{\rho}$*-Hessian Lipschitz continuous; that is,* $\|\nabla^2\varphi(x) - \nabla^2\varphi(x')\| \leq \widetilde{\rho}\|x - x'\|$ *for any* $x, x' \in \mathbb{R}^{d_x}$*, where* $\widetilde{\rho} = \mathcal{O}(\kappa^5)$

$$\widetilde{\rho} \triangleq \left(\rho + \frac{2\ell\rho + M\nu}{\mu} + \frac{2M\ell\nu + \rho\ell^2}{\mu^2} + \frac{M\ell^2\nu}{\mu^3}\right)\left(1 + \frac{\ell}{\mu}\right) \\ + \left(\frac{2\ell\rho}{\mu} + \frac{4M\rho^2 + 2\ell^2\rho}{\mu^2} + \frac{2M\ell\rho^2}{\mu^3}\right)\left(1 + \frac{\ell}{\mu}\right)^2 + \left(\frac{M\rho^2}{\mu^2} + \frac{\rho\ell}{\mu}\right)\left(1 + \frac{\ell}{\mu}\right)^3$$

We also introduce the following property for $y^*(x)$, solution to LL problem of (3):

Lemma 2. *Suppose Assumption 1 holds, then* $y^*(x)$ *is* $\tilde{\kappa} \triangleq (\widetilde{L}/\mu)$*-Lipschitz continuous; that is,* $\|y^*(x) - y^*(x')\|_2 \leq \tilde{\kappa}\,\|x - x'\|_2$ *for any* $x, x' \in \mathbb{R}^{d_x}$*.*

Similar to Condition 10 in [YLL+23, §3] for the analysis of `RAHGD`, we introduce a condition that bounds the estimation error of $y^*(w_{t,k})$ and $z^*(w_{t,k})$ after running `AGD` for sufficient number of iterates. Let $\lambda > 0$ be a regularization parameter that can grow with inverse precision, to be assigned later.

Condition 1. *Let* $w_{t,-1} = x_{t,-1}$ *and denote* $y^*(w_{t,k}) = \arg\min\ f(w_{t,k}, \cdot) + \lambda g(w_{t,k}, \cdot)$, $z^*(w_{t,k}) = \arg\min\ g(w_{t,k}, \cdot)$*. Then for some* $\sigma > 0$ *and* $t = 0, 1, 2, \ldots$*, we assume that the estimators* $y_{t,k} \in \mathbb{R}^{d_y}$ *and* $z_{t,k} \in \mathbb{R}^{d_y}$ *satisfy the conditions*

$$\|y_{t,k} - y^*(w_{t,k})\|_2 \leq \frac{\sigma}{2(1+\lambda)\ell} \qquad \text{for each } k = -1, 0, 1, 2, \ldots \tag{7}$$

and

$$\|z_{t,k} - z^*(w_{t,k})\|_2 \leq \frac{\sigma}{2\ell} \qquad \text{for each } k = -1, 0, 1, 2, \ldots \tag{8}$$

We then introduce the following gradient approximation that calibrates the inexactness of our gradient estimate

$$\widehat{\nabla}\varphi(x_k) = \nabla_x f(x_k, y_k) - \nabla^2_{xy} g(x_k, y_k) v_k$$

where

$$v_k \triangleq \left(\nabla^2_{yy} g(x_k, y_k)\right)^{-1} \nabla_y f(x_k, y_k)$$

and conclude

Lemma 3 (Inexact gradients)**.** *Suppose Assumption 1 and Condition 1 hold, then we have*

$$\|\nabla\varphi(w_k) - \widehat{\nabla}\varphi(w_k)\|_2 \leq \sigma$$

Finally as an important component of our algorithmic design, we introduce here Algorithm 1, namely Nesterov's *accelerated gradient descent* (`AGD`) for a given smooth and strongly convex objective, which achieve optimality among first-order methods in its setting. The method achieve the following *optimal rate* [N+18]:[3]

[3] One can replace this by any subroutine that achieves essentially the same optimal rate; see, e.g., [Rd17].

Algorithm 1 AGD$(h, z_0, T, \alpha, \beta)$, Nesterov's Acceleration

```
Input: objective $h(\cdot)$; initialization $z_0$; iteration number $T \geq 1$; step-size $\alpha > 0$; momentum param. $\beta \in (0,1)$
$\widetilde{z}_0 \leftarrow z_0$
for $t = 0, \ldots, T-1$ do
    $z_{t+1} \leftarrow \widetilde{z}_t - \alpha \nabla h(\widetilde{z}_t)$
    $\widetilde{z}_{t+1} \leftarrow z_{t+1} + \beta(z_{t+1} - z_t)$
end for
Output: $z_T$
```

Lemma 4. *Running Algorithm 1 on an ℓ_h-smooth and μ_h-strongly convex objective function $h(\cdot)$ with $\alpha = 1/\ell_h$ and $\beta = (\sqrt{\kappa_h} - 1)/(\sqrt{\kappa_h} + 1)$ produces an output z_T satisfying*

$$\|z_T - z^*\|_2^2 \leq (1 + \kappa_h) \left(1 - \frac{1}{\sqrt{\kappa_h}}\right)^T \|z_0 - z^*\|_2^2$$

where $z^* = \arg\min_z \ h(z)$ and $\kappa_h = \ell_h/\mu_h$ denotes the condition number of the objective h.

Alternatively, the *conjugate gradient* (CG) method was often used to further improve the rate for minimizing quadratic objective of form $\frac{1}{2}q^\top Aq - q^\top b$ where matrix $A \in \mathbb{R}^{d\times d}$ is positive definite and vector $b \in \mathbb{R}^d$ is arbitrary. The conjugate gradient method is used not in this work but heavily in [YLL+23] in designing RAHGD and PRAHGD, and we forgo its details. Under slightly different oracles the algorithm achieves an accelerated convergence rate with an improved coefficient. See [NW06] for more on the details.

2.2 Theoretical Guarantees for Accelerated Fully First-Order Methods

In this subsection, we propose the *fully first-order methods* for BiO [KKWN23, CMZ23] and draw connections between our algorithmic framework and theirs. For further analysis we recall our Assumption 1 which our theoretical result highly relies upon. Here we present some properties of function $\mathcal{L}_\lambda^*(x)$ and its connection to function $\varphi(x)$ in the following lemma [CMZ23].

Lemma 5. *Suppose Assumption 1(i)–(iv) hold and set $\lambda \geq 2\kappa$, then*

(i) $|\mathcal{L}_\lambda^*(x) - \varphi(x)| \leq \mathcal{O}(\kappa^2/\lambda)$ *for any* $x \in \mathbb{R}^{d_x}$

(ii) $\|\nabla\mathcal{L}_\lambda^*(x) - \nabla\varphi(x)\| \leq \mathcal{O}(\kappa^3/\lambda)$ *for any* $x \in \mathbb{R}^{d_x}$

(iii) $\mathcal{L}_\lambda^*(x)$ *is* L_λ*-gradient Lipschitz, where* $L_\lambda = \mathcal{O}(\kappa^3)$

If we further suppose Assumption 1(v) holds, then

(i) $\left\|\nabla^2\mathcal{L}_\lambda^*(x) - \nabla^2\varphi(x)\right\| \leq \mathcal{O}(\kappa^6/\lambda)$ *for any* $x \in \mathbb{R}^{d_x}$

(ii) $\mathcal{L}_\lambda^*(x)$ *is* ρ_λ*-Hessian Lipschitz, where* $\rho_\lambda = \mathcal{O}(\kappa^5)$

The detailed expression for error controls $\|\nabla\mathcal{L}_\lambda^*(x) - \nabla\varphi(x)\|_2$, L_λ, $|\mathcal{L}_\lambda^*(x) - \varphi(x)|$, $\left\|\nabla^2\mathcal{L}_\lambda^*(x) - \nabla^2\varphi(x)\right\|_2$ and ρ_λ can be found in §B.2.

We propose detailed *(Perturbed) Restarted Accelerated Fully First-order methods for Bilevel Approximation* Algorithm 2, or (P)RAF^{2}BA for short. The theoretical guarantees of this algorithm is presented as follows:

Algorithm 2 (Perturbed) Restarted Accelerated F^{2}BA, (P)RAF^{2}BA

```
Input: initial vector $x_{0,0}$; step-size $\eta > 0$; momentum parameter $\theta \in (0,1)$; parameters $\alpha, \alpha' > 0$, $\beta, \beta' \in (0,1)$, $\{T_{t,k}\}$, $\{T'_{t,k}\}$ of AGD; iteration threshold $K \geq 1$; parameter $B$ for triggering restarting; perturbation radius $r > 0$; option Perturbation $\in \{0,1\}$
$k \leftarrow 0,\ t \leftarrow 0,\ x_{0,-1} \leftarrow x_{0,0}$
$y_{0,-1} \leftarrow \texttt{AGD}(f(x_{0,-1},\cdot) + \lambda g(x_{0,-1},\,\cdot\,), 0, T'_{0,-1}, \alpha', \beta')$
$z_{0,-1} \leftarrow \texttt{AGD}(g(x_{0,-1},\,\cdot\,), 0, T_{0,-1}, \alpha, \beta)$
while $k < K$ do
    $w_{t,k} \leftarrow x_{t,k} + (1-\theta)(x_{t,k} - x_{t,k-1})$
    $z_{t,k} \leftarrow \texttt{AGD}(g(w_{t,k},\,\cdot\,), z_{t,k-1}, T_{t,k}, \alpha, \beta)$
    $y_{t,k} \leftarrow \texttt{AGD}(f(w_{t,k},\cdot) + \lambda g(w_{t,k},\,\cdot\,), y_{t,k-1}, T'_{t,k}, \alpha', \beta')$
    $u_{t,k} \leftarrow \nabla_x f(w_{t,k}, y_{t,k}) + \lambda(\nabla_x g(w_{t,k}, y_{t,k}) - \nabla_x g(w_{t,k}, z_{t,k}))$
    $x_{t,k+1} \leftarrow w_{t,k} - \eta u_{t,k}$
    $k \leftarrow k+1$
    if $k \sum_{i=0}^{k-1} \|x_{t,i+1} - x_{t,i}\|^2 > B^2$ then
        if Perturbation $= 0$ then
            $x_{t+1,0} \leftarrow x_{t,k}$
        else
            $x_{t+1,0} \leftarrow x_{t,k} + \xi_{t,k}$  with  $\xi_{t,k} \sim \mathrm{Unif}(\mathbb{B}_r)$
        end if
        $x_{t+1,-1} \leftarrow x_{t+1,0}$
        $y_{t+1,-1} \leftarrow \texttt{AGD}(f(x_{t+1,-1},\cdot) + \lambda g(x_{t+1,-1},\,\cdot\,), 0, T'_{t+1,-1}, \alpha', \beta')$
        $z_{t+1,-1} \leftarrow \texttt{AGD}(g(x_{t+1,-1},\,\cdot\,), 0, T_{t+1,-1}, \alpha, \beta)$
        $k \leftarrow 0,\ t \leftarrow t+1$
    end if
end while
$K_0 \leftarrow \arg\min_{\lfloor \frac{K}{2} \rfloor \leq k \leq K-1} \ \|x_{t,k+1} - x_{t,k}\|_2$
Output: $\widehat{w} \leftarrow \frac{1}{K_0+1} \sum_{k=0}^{K_0} w_{t,k}$
```

Theorem 2 (RAF^{2}BA finding ϵ-FOSP)**.** *Suppose Assumptions 1 holds. Let $\Delta = \varphi(x_{\mathrm{int}}) - \min_{x\in\mathbb{R}^{d_x}} \ \varphi(x)$, $\kappa' = (\lambda+1)\ell/(\lambda\mu - \ell)$, and*

$$\eta = \frac{1}{4L_\lambda} \qquad B = \sqrt{\frac{\epsilon}{\rho_\lambda}} \qquad \theta = (\rho_\lambda \epsilon \eta^2)^{1/4} \qquad K = \frac{1}{\theta} \qquad \alpha = \frac{1}{\ell} \qquad \beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$$

$$\lambda = \Theta\left(\max\{\kappa^2/\Delta,\ \kappa^3/\epsilon\}\right) \qquad \alpha' = \frac{1}{(\lambda+1)\ell} \qquad \beta' = \frac{\sqrt{\kappa'}-1}{\sqrt{\kappa'}+1} \qquad \sigma = \epsilon^2$$

and assume that $\mathcal{O}(\epsilon) \leq L_\lambda^2/\rho_\lambda$. Then our RAF^{2}BA (Algorithm 2) can find an $\mathcal{O}(\epsilon)$-first-order stationary point of $\varphi(x)$. Additionally, the oracle complexities satisfy $Gc(f,\epsilon) = Gc(g,\epsilon) = \widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$.

When κ reduces to 1 the algorithm can be adapted to solve the single-level nonconvex minimization problem, matching the state-of-the-art complexity [CDHS18, AAZB$^+$17, CDHS17, JNJ18, LL23]. The best-known lower bound in this context is $\Omega(\epsilon^{-1.714})$ [CDHS21]. Analogous to accelerating inexact hypergradient method as in [YLL$^+$23], we have the following perturbed version to hold:

Table 1. Comparison table for nonconvex-strongly-convex BiO algorithms, finding approximate FOSP (top six rows) and SOSP (bottom four rows)

Complexities Algorithm	$Gc(f,\epsilon)$	$Gc(g,\epsilon)$	$JV(g,\epsilon)$	$HV(g,\epsilon)$
AID-BiO [JYL21, GW18]	$\mathcal{O}(\kappa^3\epsilon^{-2})$	$\mathcal{O}(\kappa^4\epsilon^{-2})$	$\mathcal{O}(\kappa^3\epsilon^{-2})$	$\mathcal{O}(\kappa^{3.5}\epsilon^{-2})$
ITD-BiO [JYL21]	$\mathcal{O}(\kappa^3\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$
F^{2}BA [CMZ23, KKWN23]	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	0	0
`RAHGD` [YLL+23]	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$
`RAF`^{2}`BA` (this work)	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$	0	0
Perturbed AID [HJML22]	$\widetilde{\mathcal{O}}(\kappa^3\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^3\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^{3.5}\epsilon^{-2})$
Perturbed F^{2}BA [CMZ23]	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa^4\epsilon^{-2})$	0	0
`PRAHGD` [YLL+23]	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$
`PRAF`^{2}`BA` (this work)	$\widetilde{\mathcal{O}}(\kappa^{2.75}\epsilon^{-1.75})$	$\widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$	0	0

• $Gc(f,\epsilon)$ and $Gc(g,\epsilon)$: gradient query complexity of f and g • $JV(g,\epsilon)$: Jacobian-vector-product query complexity of g • $HV(g,\epsilon)$: Hessian-vector product query complexity of g • $\widetilde{\mathcal{O}}(\cdot)$ omits a polylogarithmic factor in problem-dependent parameters • κ denotes the condition number of LL objective

Theorem 3 (`PRAF`^{2}`BA` finding $(\epsilon, \mathcal{O}(\kappa^{2.5}\sqrt{\epsilon}))$-SOSP)**.** *Suppose Assumption 1 holds. Let* $\Delta = \varphi(x_{\text{int}}) - \min_{x\in\mathbb{R}^{d_x}} \varphi(x)$*,* $\kappa' = (\lambda+1)\ell/(\lambda\mu - \ell)$*, and*

$$\chi = \mathcal{O}\left(\log\frac{d_x}{\zeta\epsilon}\right) \qquad \eta = \frac{1}{4L_\lambda} \qquad K = \frac{2\chi}{\theta} \qquad B = \frac{1}{288\chi^2}\sqrt{\frac{\epsilon}{\rho_\lambda}} \qquad \theta = \frac{1}{2}(\rho_\lambda\epsilon\eta^2)^{1/4}$$

$$\alpha = \frac{1}{\ell} \qquad \beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \qquad \alpha' = \frac{1}{(\lambda+1)\ell} \qquad \beta' = \frac{\sqrt{\kappa'}-1}{\sqrt{\kappa'}+1}$$

$$r = \min\left\{\frac{L_\lambda B^2}{4C}, \frac{B+B^2}{\sqrt{2}}, \frac{\theta B}{20K}, \sqrt{\frac{\theta B^2}{2K}}\right\} \qquad \lambda = \Theta\left(\max\left\{\frac{\kappa^2}{\Delta}, \frac{\kappa^3}{\epsilon}, \frac{\kappa^6}{\sqrt{\epsilon}}\right\}\right) \qquad \sigma = \min\left\{\frac{\rho_\lambda B\zeta r\theta}{2\sqrt{d_x}}, \epsilon^2\right\}$$

for some positive constant C and assume that $\epsilon \le L_\lambda^2/\rho_\lambda$. Then our ***PRAF***^{2}***BA*** *(Algorithm 2 with* ***Perturbation*** $= 1$*) can find an $(\mathcal{O}(\epsilon), \mathcal{O}(\kappa^{2.5}\sqrt{\epsilon}))$-second-order stationary point of $\varphi(x)$ with probability at least $1-\zeta$. Additionally, the oracle complexities satisfy $Gc(f,\epsilon) = Gc(g,\epsilon) = \widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$.*

The presented oracle-call query complexities match the state-of-the-art and are almost identical to those in Theorem 2, differing only by a polylogarithmic factor. This indicates that the perturbed version incurs essentially no additional cost while enabling the avoidance of saddle points. In comparison with [YLL+23], the presented query complexities does not invoke any Hessian-vector-product or Jacobian-vector-product queries, and is hence *fully first-order*. A detailed comparison is listed in Table 1.

2.3 Proof of Theorem 2

From Lemma 5, setting $\lambda = \Theta\left(\max\{\kappa^2/\Delta,\ \kappa^3/\epsilon\}\right)$ leads to

- $\|\nabla\varphi(x) - \nabla\mathcal{L}_\lambda^*(x)\| \le \mathcal{O}(\epsilon)$, for any $x \in \mathbb{R}^{d_x}$
- $\mathcal{L}_\lambda^*(x_{\text{int}}) - \min_{x\in\mathbb{R}^{d_x}} \mathcal{L}_\lambda^*(x) \le \mathcal{O}(\Delta)$

Thus, we only need to prove that RAF^{2}BA (in Algorithm 2) can find an ϵ-first-order stationary point of $\mathcal{L}_\lambda^*(x)$ within the desired complexity.

Under Condition 1 and Assumption 1 we have the following lemma.

Lemma 6. *Suppose Assumption 1 and Condition 1 hold, then for each $k = -1, 0, 1, \dots,$ and $t = 0, 1, 2, \dots,$ we have*

$$\|u_{t,k} - \nabla \mathcal{L}_\lambda^*(w_{t,k})\|_2 \leq \sigma$$

where $u_{t,k}$ is defined in Line 9 in Algorithm 2.

Proof of Lemma 6. Note that

$$u_{t,k} = \nabla_x f(w_{t,k}, y_{t,k}) + \lambda(\nabla_x g(w_{t,k}, y_{t,k}) - \nabla_x g(w_{t,k}, z_{t,k}))$$

and

$$\nabla \mathcal{L}_\lambda^*(w_{t,k}) = \nabla_x f(w_{t,k}, y^*(w_{t,k})) + \lambda(\nabla_x g(w_{t,k}, y^*(w_{t,k})) - \nabla_x g(w_{t,k}, z^*(w_{t,k}))$$

Then from Condition 1 and the Lipschitz continuity of gradient of f and g, we have

$$\|u_{t,k} - \nabla \mathcal{L}_\lambda^*(w_{t,k})\| \leq (1+\lambda)\ell \cdot \frac{\sigma}{2(1+\lambda)\ell} + \ell \cdot \frac{\sigma}{2\ell} = \sigma$$

proving the lemma. □

Note that the only difference of Algorithm 2 and the PRAHGD algorithm proposed in [YLL+23, Algorithm 2] lies on the constructions of the inexact gradient of the objective functions, i.e., $\nabla \mathcal{L}_\lambda^*(w_{t,k}) \approx u_{t,k} = \nabla_x f(w_{t,k}, y_{t,k}) + \lambda\left(\nabla_x g(w_{t,k}, y_{t,k}) - \nabla_x g(w_{t,k}, z_{t,k})\right)$ for Algorithm 2 and $\nabla \varphi(w_{t,k}) \approx u_{t,k} = \nabla_x f(w_{t,k}, y_{t,k}) - \nabla_{xy}^2 g(w_{t,k}, y_{t,k}) v_{t,k}$ for PRAHGD. Thus, we can directly follow the proof of [YLL+23, Theorem 14] by replacing $\varphi(x)$ by $\mathcal{L}_\lambda^*(x)$ and achieve the following result:

Theorem 4. *Suppose that Assumptions 1 and Condition 1 hold. Denote $\Delta_\lambda = \mathcal{L}_\lambda^*(x_{\text{int}}) - \min_{x \in \mathbb{R}^{d_x}} \mathcal{L}_\lambda^*(x)$ and $\kappa' = (\lambda+1)\ell/(\lambda\mu - \ell)$ (recall our choice of $\lambda \geq 2\kappa$). Let*

$$\eta = \frac{1}{4L_\lambda} \qquad B = \sqrt{\frac{\epsilon}{\rho_\lambda}} \qquad \theta = 4(\rho_\lambda \epsilon \eta^2)^{1/4} \qquad K = \frac{1}{\theta} \qquad \alpha = \frac{1}{\ell} \qquad \beta = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}$$

$$\alpha' = \frac{1}{(\lambda+1)\ell} \qquad \beta' = \frac{\sqrt{\kappa'} - 1}{\sqrt{\kappa'} + 1} \qquad \sigma = \epsilon^2$$

and assume that $\epsilon \leq L_\lambda^2/\rho_\lambda$. Then our RAF^{2}BA in Algorithm 2 terminates within $\mathcal{O}(\Delta_\lambda L_\lambda^{0.5} \rho_\lambda^{0.25} \epsilon^{-1.75})$ iterates, outputting $\widehat{w}$ satisfying $\|\nabla \mathcal{L}_\lambda^*(\widehat{w})\| \leq 83\epsilon$.

Now we consider the overall inner loop iteration number from the step of AGD to achieve $z_{t,k}$ in the algorithm. Following the proof of [YLL+23, Lemma 31] (§D.2 therein), we achieve the upper bound of $\|z^*(w_{t,-1})\|_2 \leq \widehat{C}_z$ as follows.

Lemma 7. *Consider the setting of Theorem 4, and we run Algorithm 2, then we have*

$$\|z^*(w_{t,-1})\| \leq \widehat{C}_z$$

for any $t > 0$ and some constant $C > 0$, where $\widehat{C}_z = \|z^*(x_{0,0})\|_2 + (2B + \eta\sigma + \eta C)\kappa \Delta_\lambda \sqrt{\rho_\lambda} \epsilon^{-3/2}$.

Taking

$$T_{t,k} = \begin{cases} \left\lceil 2\sqrt{\kappa}\log\left(\frac{2\ell\sqrt{\kappa+1}}{\sigma}\widehat{C}_z\right)\right\rceil & k=-1 \\ \left\lceil 2\sqrt{\kappa}\log\left(\frac{2\ell\sqrt{\kappa+1}}{\sigma}\left(\frac{\sigma}{2\ell}+2\kappa B\right)\right)\right\rceil & k\geq 0 \end{cases} \tag{9}$$

for Algorithm 2, we can use induction to show Lemma 7 and (8) in Condition 1 hold, which is similar to the analysis in [YLL+23, §D.2].

Finally, we consider the overall inner loop iteration number from the step of AGD to achieve $y_{t,k}$ in the algorithm. Following the proof of [YLL+23, Lemma 31] (§D.2 therein), we achieve the upper bound of $\|y^*(w_{t,-1})\|_2 \leq \widehat{C}_y$ as follows.

Lemma 8. *Consider the setting of Theorem 4, and we run Algorithm 2, then we have*

$$\|y^*(w_{t,-1})\| \leq \widehat{C}_y$$

for any $t = 0,1,2,\ldots$ *and some constant* $C > 0$*, where* $\widehat{C}_y = \|y^*(x_{0,0})\| + (2B+\eta\sigma+\eta C)\kappa'\Delta_\lambda\sqrt{\rho_\lambda}\epsilon^{-3/2}$.

Notice that the condition number of $f(x,\cdot)+\lambda g(x,\cdot)$ is $\kappa' = (\lambda+1)\ell/(\lambda\mu-\ell) = \mathcal{O}(\kappa)$ for any $x \in \mathbb{R}^{d_x}$. Analogizing the setting of $T_{t,k}$, we take

$$T'_{t,k} = \begin{cases} \left\lceil 2\sqrt{\kappa'}\log\left(\frac{2(1+\lambda)\ell\sqrt{\kappa'+1}}{\sigma}\widehat{C}_y\right)\right\rceil & k=-1 \\ \left\lceil 2\sqrt{\kappa'}\log\left(\frac{2(\lambda+1)\ell\sqrt{\kappa'+1}}{\sigma}\left(\frac{\sigma}{2(\lambda+1)\ell}+2\kappa' B\right)\right)\right\rceil & k\geq 0 \end{cases} \tag{10}$$

for Algorithm 2. We can also use induction to show Lemma 8 and (7) in Condition 1 hold, which is similar to the analysis in [YLL+23, §D.2].

Combining Theorem 4 with the above settings of $T_{t,k}$ and $T'_{t,k}$, we conclude that our RAF^{2}BA can find an ϵ-first-order stationary point of $\mathcal{L}^*_\lambda(x)$ (also an $\mathcal{O}(\epsilon)$-first-order stationary point of $\varphi(x)$) within oracle complexities $Gc(f,\epsilon) = Gc(g,\epsilon) = \widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$, which is similar to the proof of [YLL+23, Corollary 15] (§D.3 therein).

2.4 Proof of Theorem 3

From Lemma 5, setting $\lambda = \Theta\left(\max\{\kappa^2/\Delta,\ \kappa^3/\epsilon,\ \kappa^6/\sqrt{\epsilon}\}\right)$ leads to

- $\|\nabla\varphi(x) - \nabla\mathcal{L}^*_\lambda(x)\| \leq \mathcal{O}(\epsilon)$, for any $x \in \mathbb{R}^{d_x}$
- $\left\|\nabla^2\varphi(x) - \nabla^2\mathcal{L}^*_\lambda(x)\right\| \leq \mathcal{O}(\sqrt{\epsilon})$, for any $x \in \mathbb{R}^{d_x}$
- $\mathcal{L}^*_\lambda(x_{\text{int}}) - \min_{x\in\mathbb{R}^{d_x}}\ \mathcal{L}^*_\lambda(x) \leq \mathcal{O}(\Delta)$

Now all we need is to show that our PRAF^{2}BA can find an $(\epsilon, O(\kappa^{2.5}\sqrt{\epsilon}))$-second-order stationary point of $\mathcal{L}^*_\lambda(x)$ within the desired complexity.

Following the proof of [YLL+23, Theorem 16], we have the following theorem.

Theorem 5. *Suppose that Assumption 1 and Condition 1 hold. We denote* $\Delta_\lambda = \mathcal{L}^*_\lambda(x_{\text{int}}) - \min_{x\in\mathbb{R}^{d_x}}\mathcal{L}^*_\lambda(x)$ *and* $\kappa' = (\lambda+1)\ell/(\lambda\mu-\ell)$ *and let*

$$\chi = \mathcal{O}\left(\log\frac{d_x}{\zeta\epsilon}\right) \quad \eta = \frac{1}{4L_\lambda} \quad K = \frac{2\chi}{\theta} \quad B = \frac{1}{288\chi^2}\sqrt{\frac{\epsilon}{\rho_\lambda}} \quad \theta = \frac{1}{2}(\rho_\lambda\epsilon\eta^2)^{1/4} \quad \sigma = \min\left\{\frac{\rho_\lambda B\zeta r\theta}{2\sqrt{d_x}}, \epsilon^2\right\}$$

$$\alpha = \frac{1}{\ell} \quad \beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \quad \alpha' = \frac{1}{(\lambda+1)\ell} \quad \beta' = \frac{\sqrt{\kappa'}-1}{\sqrt{\kappa'}+1} \quad r = \min\left\{\frac{L_\lambda B^2}{4C}, \frac{B+B^2}{\sqrt{2}}, \frac{\theta B}{20K}, \sqrt{\frac{\theta B^2}{2K}}\right\}$$

for some positive constant C, where we assume that $\epsilon \le L_\lambda^2/\rho_\lambda$. Then `PRAF`^{2}`BA` *in Algorithm 2 terminates in at most $\mathcal{O}\big(\Delta_\lambda L_\lambda^{0.5}\rho_\lambda^{0.25}\chi^6 \cdot \epsilon^{-1.75}\big)$ iterations and the output satisfies $\|\nabla\mathcal{L}_\lambda^*(\widehat{w})\| \le \epsilon$ and $\lambda_{\min}(\nabla^2\mathcal{L}_\lambda^*(\widehat{w})) \ge -1.011\sqrt{\rho_\lambda\epsilon}$ with probability at least $1-\zeta$.*

Now we set parameters $T_{t,k}$ and $T'_{t,k}$ in a similar way to the counterparts in [YLL+23, §E.2] and §2.3, that is,

$$T_{t,k} = \begin{cases} \left\lceil 2\sqrt{\kappa}\log\left(\frac{2\ell\sqrt{\kappa+1}}{\sigma}\widetilde{C}_z\right)\right\rceil & k=-1 \\ \left\lceil 2\sqrt{\kappa}\log\left(\frac{2\ell\sqrt{\kappa+1}}{\sigma}\left(\frac{\sigma}{2\ell}+2\kappa B\right)\right)\right\rceil & k\ge 0 \end{cases} \tag{11}$$

and

$$T'_{t,k} = \begin{cases} \left\lceil 2\sqrt{\kappa'}\log\left(\frac{2(1+\lambda)\ell\sqrt{\kappa'+1}}{\sigma}\widetilde{C}_y\right)\right\rceil & k=-1 \\ \left\lceil 2\sqrt{\kappa'}\log\left(\frac{2(\lambda+1)\ell\sqrt{\kappa'+1}}{\sigma}\left(\frac{\sigma}{2(\lambda+1)\ell}+2\kappa' B\right)\right)\right\rceil & k\ge 0 \end{cases} \tag{12}$$

where

$$\widetilde{C}_z = \|z^*(x_{0,0})\| + (2B + B^2 + \eta\sigma + \eta C)\kappa\Delta_\lambda\sqrt{\rho_\lambda}\epsilon^{-3/2}$$

and

$$\widetilde{C}_y = \|y^*(x_{0,0})\| + (2B + B^2 + \eta\sigma + \eta C)\kappa'\Delta_\lambda\sqrt{\rho_\lambda}\epsilon^{-3/2}$$

We can also use induction to prove that Condition 1 will hold when we choose $T_{t,k}$ and $T'_{t,k}$ as set in (11) and (12).

Combining Theorem 5 with the above settings of $T_{t,k}$ and $T'_{t,k}$, we conclude that our `PRAF`^{2}`BA` can find an $(\epsilon, \kappa^{2.5}\mathcal{O}(\sqrt{\epsilon}))$-second-order stationary point of $\mathcal{L}_\lambda^*(x)$ (also an $(\mathcal{O}(\epsilon), \kappa^{2.5}\mathcal{O}(\sqrt{\epsilon}))$-second-order stationary point of $\varphi(x)$) within oracle complexities $Gc(f,\epsilon) = Gc(g,\epsilon) = \widetilde{\mathcal{O}}(\kappa^{3.25}\epsilon^{-1.75})$, which is similar to the proof of [YLL+23, Corollary 18] (§D therein).

3 `PRAF`^{2}`BA` for Accelerating NCSC Minimax Optimization

This section applies the ideas of `PRAF`^{2}`BA` to find an *approximate second-order stationary point* in the *nonconvex-concave* minimax optimization problem of the form

$$\min_{x\in\mathbb{R}^{d_x}} \left\{\bar{\varphi}(x) \triangleq \max_{y\in\mathbb{R}^{d_y}} \bar{f}(x,y)\right\} \tag{13}$$

where the minimax objective $\bar{f}(x,y)$ is (strongly) concave in y but possibly nonconvex in x. As is discussed in [YLL+23, §B], minimax problems of form (13) can be regarded as a special case of our BiO problem (3) with $f(x,y) = \bar{f}(x,y)$ and $g(x,y) = -\bar{f}(x,y)$. We first show in the upcoming Fact 6 that the derivatives of our minimax objective enjoy tighter Lipschitz constants than the general BiO problem, as is established in §2.1:

Fact 6 ([LLC22, YLL+23]). *Let $\bar{f}(x,y)$ be ℓ-smooth, ρ-Hessian Lipschitz continuous with respect to x and y and μ-strongly concave in y but possibly nonconvex in x. Then the hyper-objective $\bar{\varphi}(x)$ is $(\kappa+1)\ell$-smooth and has $(4\sqrt{2}\kappa^3\rho)$-Lipschitz continuous Hessians.*

In comparison with the task of finding approximate second-order stationary point using BiO, one observes from Fact 6 that the κ-dependency in the negated Hessian precision is improved from $\kappa^{2.5}$ to $\kappa^{1.5}$, and our goal is to find a (more stringent) $\big(\epsilon, O(\kappa^{1.5}\sqrt{\epsilon})\big)$-second-order stationary point of $\bar{\varphi}(x)$.

Algorithm 3 Perturbed Restarted Accelerated Gradient Descent Ascent, `PRAGDA`

```
1: Input: initial vector $x_{0,0}$; step-size $\eta > 0$; momentum param. $\theta \in (0,1)$; params. $\alpha > 0, \beta \in (0,1), \{T_{t,k}\}$ of AGD; iteration threshold $K \geq 1$; param. $B$ for triggering restarting; perturbation radius $r > 0$
2: $k \leftarrow 0,\ t \leftarrow 0,\ x_{0,-1} \leftarrow x_{0,0}$
3: $y_{0,-1} \leftarrow \mathtt{AGD}(-\bar{f}(x_{0,-1},\ \cdot\ ), 0, T_{0,-1}, \alpha, \beta)$
4: while $k < K$ do
5:     $w_{t,k} \leftarrow x_{t,k} + (1-\theta)(x_{t,k} - x_{t,k-1})$
6:     $y_{t,k} \leftarrow \mathtt{AGD}(-\bar{f}(w_{t,k},\ \cdot\ ), y_{t,k-1}, T_{t,k}, \alpha, \beta)$
7:     $x_{t,k+1} \leftarrow w_{t,k} - \eta \nabla_x \bar{f}(w_{t,k}, y_{t,k})$
8:     $k \leftarrow k + 1$
9:     if $k \sum_{i=0}^{k-1} \|x_{t,i+1} - x_{t,i}\|^2 > B^2$ then
10:        $x_{t+1,0} \leftarrow x_{t,k} + \xi_{t,k}$ with $\xi_{t,k} \sim \mathrm{Unif}(\mathbb{B}_r)$
11:        $x_{t+1,-1} \leftarrow x_{t+1,0}$
12:        $k \leftarrow 0,\ t \leftarrow t + 1$
13:        $y_{t,-1} \leftarrow \mathtt{AGD}(-\bar{f}(x_{t,-1},\ \cdot\ ), 0, T_{t,-1}, \alpha, \beta)$
14:    end if
15: end while
16: $K_0 \leftarrow \arg\min_{\lfloor \frac{K}{2} \rfloor \leq k \leq K-1} \|x_{t,k+1} - x_{t,k}\|$
17: Output: $\widehat{w} \leftarrow \frac{1}{K_0+1} \sum_{k=0}^{K_0} w_{t,k}$
```

Connection between `PRAF`^{2}`BA` and the *perturbed restarted accelerated gradient descent ascent.* We recap the *perturbed restarted accelerated gradient descent ascent* (`PRAGDA`) introduced by [YLL+23] (Algorithm 3 therein) as a special case of their proposed algorithm, `PRAHGD`. Algorithmic details are provided in Algorithm 3. As we will point out immediately, this is exactly our Algorithm 2 applied to minimax problem (13).

When applying to the minimax problem (13), the procedures of Algorithm 2 (with `Perturbation` = 1) and Algorithm 3 are *identical* with the appropriate parameters setup. We observe that since $\lambda > 1$, the regularized objective $\mathcal{L}_\lambda^*(x)$ is exactly equal to the objective function $\bar{\varphi}(x)$ in minimax problem (13). Indeed, function $\mathcal{L}_\lambda^*(x)$ can be written as

$$
\begin{aligned}
\mathcal{L}_\lambda^*(x) &= \min_{y \in \mathbb{R}^{d_y}} \left(f(x,y) + \lambda \left(g(x,y) - \min_{z \in \mathbb{R}^{d_y}} g(x,z) \right) \right) \\
&= \min_{y \in \mathbb{R}^{d_y}} \left(\bar{f}(x,y) + \lambda \left(-\bar{f}(x,y) - \min_{z \in \mathbb{R}^{d_y}} -\bar{f}(x,z) \right) \right) = \min_{y \in \mathbb{R}^{d_y}} \left((1-\lambda)\bar{f}(x,y) + \lambda \max_{z \in \mathbb{R}^{d_y}} \bar{f}(x,z) \right) \\
&= (1-\lambda) \max_{y \in \mathbb{R}^{d_y}} \bar{f}(x,y) + \lambda \max_{z \in \mathbb{R}^{d_y}} \bar{f}(x,z) = \max_{y \in \mathbb{R}^{d_y}} \bar{f}(x,y)
\end{aligned}
$$

which reduces to the objective function $\bar{\varphi}(x)$ in the minimax problem (13).

Now, careful examination of the algorithm procedures indicates that applying Algorithm 2 to minimizing $\mathcal{L}_\lambda^*(x)$ with $\alpha = \alpha'$ and $\beta = \beta'$ implies that the $y_{t,k} = z_{t,k}$ always holds, since the sequences $\{y_{t,k}\}$ and $\{z_{t,k}\}$ correspond to the iterations for problems $\min_{y \in \mathbb{R}^{d_y}} \ - f(w_{t,k}, y)$ and $\min_{y \in \mathbb{R}^{d_y}} \ -(\lambda - 1) f(w_{t,k}, y)$, respectively. Hence, Lines 7—8 of Algorithm 2 is identical to Line 7 of Algorithm 3 when $\eta = \eta_x$, proving the equivalence.

Therefore under this setup, utilizing Fact 6 we can take $\widetilde{L} = (\kappa + 1)\ell$ and $\widetilde{\rho} = 4\sqrt{2}\kappa^3\rho$ to conclude an improved oracle complexity upper bounds for finding second-order stationary points

Table 2. Comparisons of gradient query complexity for finding approximate SOSP in NCSC minimax optimization algorithms

Complexities Algorithm	$Gc(\bar{f}, \epsilon)$	$HV(\bar{f}, \epsilon)$	$JV(\bar{f}, \epsilon)$
IMCN [LLC22]	$\widetilde{\mathcal{O}}(\kappa^2\epsilon^{-1.5})$	$\widetilde{\mathcal{O}}(\kappa^{1.5}\epsilon^{-2})$	$\widetilde{\mathcal{O}}(\kappa\epsilon^{-2})$
`PRAGDA` ([YLL+23], this work)	$\widetilde{\mathcal{O}}(\kappa^{1.75}\epsilon^{-1.75})$	0	0

for this particular problem, indicated by the following statement:

Theorem 7 (Oracle complexity of `PRAF`^{2}`BA` for accelerating minimax optimization)**.** *For solving* (13) *under the settings of Fact 6, Algorithm 2 reduces to Algorithm 3 which outputs an* $\big(\epsilon, \mathcal{O}(\kappa^{1.5}\sqrt{\epsilon})\big)$*-second-order stationary point of* $\bar{\varphi}(x)$ *in* (13) *within* $\widetilde{\mathcal{O}}(\kappa^{1.75}\epsilon^{-1.75})$ *gradient query complexity of* $\bar{f}(x, y)$*.*

The proof of Theorem 7 is straightforward using the above equivalence, `PRAHGD` complexity result as in Theorem 16, Proposition 17 of [YLL+23], and also Fact 6. As is discussed in [YLL+23], this oracle query complexity achieves the state-of-the-art in this setting; see details in Table 2.

4 Optimality and Stationarity in Bilevel Optimization without LLSC

In this section we aim to find stationary points of the hyper-objective function, investigating when LL functions lack the typical strong convexity assumption. First, we will illustrate the intractability is mainly caused by undesirable *flatness*, and we identify two regularity conditions of the LL problems that are sufficient to provably confer tractability to BiO with only LL convexity: *the gradient dominance condition* (Assumption 4.1), and *the weak sharp minimum condition* (Assumption 4.2).

Then we present hardness results illustrating that BiO for general convex LL functions but without LLSC, for both finding an LL optimal solution and a UL stationary point, is intractable to solve [§4.2].[4] In particular

- We show that $\varphi(x)$ is not computable in finite iterations by proving a lower bound in Proposition 4.4 for general convex functions, and also in Proposition 4.5 for nonsmooth convex LL functions. [§4.2.1]

- We give a pair of $f(x, y)$ and $g(x, y)$ in Example 4.2 such that the resulting hyper-objective $\varphi(x)$ is discontinuous and thus intractable to optimize, and prove this generally holds in both ways in Proposition 4.6. [§4.2.2]

Finally, under the introduced regularity conditions, we propose novel algorithms, namely the *Inexact Gradient-Free Method* (`IGFM`), which uses the *Switching Gradient Method* (`SGM`) as an efficient sub-routine, to find an LL optimal solution and a UL stationary point as well as an approximate stationary point of the hyper-objective in polynomial time, with non-asymptotic convergence guarantees:

[4] As the readers will see in §4.2, the construction of the hard instances in the lower bound results relies on the fact that a general convex LL function can be arbitrarily *flat*.

Table 3. An overview of the theoretical results for BiO without LLSC, which is generally intractable but becomes tractable when the LL function satisfies either the gradient dominance or the weak sharp minimum condition

Assumption on LL function	LL Optimality	UL Stationarity	Reference
Strongly convex	Tractable	Tractable	Known result
Convex with dominant gradients	Tractable	Tractable	Proved by this work
Convex with weak sharp minimum	Tractable	Tractable	Proved by this work
Only convex	Intractable	Intractable	Proved by this work

- **Finding an LL Optimal Solution.** We show that both conditions fall into a general class of the Hölderian error bound condition under which we propose the *Switching Gradient Method* (`SGM`, Algorithm 4) to overcome the difficulty of multiple LL minima and find an LL optimal solution in polynomial time (Theorem 4.1) [§4.3.1].

- **Finding a UL Stationary Point.** Under the Lipschitz continuity of $\varphi(x)$, we then propose the *Inexact Gradient-Free Method* (`IGFM`, Algorithm 5) that can provably converge to a UL stationary point—a Goldstein stationary point [ZLJ+20] of the hyper-objective—by incorporating `SGM` as an efficient sub-routine [§4.3.2].[5]

§4.1 first identify several regularity conditions of the LL problems that can provably confer tractability. In §4.2 we present hardness results showing that BiO for general convex LL functions is intractable to solve. Finally in §4.3 we propose the Inexact Gradient-Free Method (`IGFM`)—which uses the Switching Gradient Method (`SGM`) as an efficient sub-routine—to find an approximate stationary point of the hyper-objective in polynomial time. Theoretical proofs and miscellaneous results are delegated to §4.4, §4.5, §C and §D.

4.1 Sufficient Conditions for Tractability

In this subsection, we provide conditions that are sufficient for tractability. §4.1.1 introduces the optimality conditions for BiO without LLSC used in this subsection. §4.1.2 introduces two assumptions corresponding to different degrees of sharpness of LL functions, which is essential to ensure the tractability of BiO.

4.1.1 The Optimality Conditions

Firstly, we recall the definition of the optimistic optimal solution [DKK06], which is a standard optimality condition for the hyper-objective reformulation.

Definition 4.1 (Locally optimistic optimality)**.** *A pair of point (x^*, y^*) is called a* locally optimistic optimal solution *to Problem (1) if $y^* \in Y^*(x^*)$ and there exists $\delta > 0$ such that we have $\varphi(x^*) \leq \varphi(x)$ and $f(x^*, y^*) \leq f(x^*, y)$ for all $(x, y) \in \mathbb{B}_\delta(x^*, y^*)$. It is called a globally optimistic optimal solution if we can let $\delta \to \infty$.*

A *globally optimistic optimal solution* is an exact solution to Problem (1), but its computation is NP-hard since $\varphi(x)$ is generally nonconvex [DDG+22]. A common relaxation is to find a locally optimistic optimal solution, for which we can derive the following necessary conditions.

[5] In fact, we will prove that both conditions imply the Lipschitz continuity of the solution mapping $Y^*(x)$, which is proved to be both sufficient and necessary for the Lipschitz continuity of $\varphi(x)$ by Proposition 4.6.

Proposition 4.1. *Suppose $f(x,\cdot)$ and $g(x,\cdot)$ are convex, and $\varphi(x)$ is locally Lipschitz. Then for any locally optimistic optimal solution (x^*,y^*), we have $\partial\varphi(x^*)=0$, $f(x^*,y^*)=\varphi(x^*)$ and $g(x^*,y^*)=g^*(x^*)$.*

It motivates us to use the following criteria for non-asymptotic analysis

Definition 4.2 (UL Stationarity)**.** *Suppose $\varphi(x)$ is locally Lipschitz. We call $\widehat{x}$ a (δ,ε)-*UL stationary point *if it is a (δ,ε)-Goldstein stationary point of $\varphi(x)$.*

Definition 4.3 (LL Optimality)**.** *Fix an x. Suppose $f(x,\cdot)$ and $g(x,\cdot)$ are convex. We call $\widehat{y}$ a (ζ_f,ζ_g)-*LL optimal solution *if we have $|f(x,\widehat{y})-\varphi(x)|\le\zeta_f$ and $g(x,\widehat{y})-g^*(x)\le\zeta_g$.*

The main focus of this section is to discuss when and how one can design a polynomial time algorithm to achieve the above goals for any given positive precision $\delta,\varepsilon,\zeta_f,\zeta_g$.

Remark 4.1. *In Definition 4.2, we assume that $\varphi(x)$ is locally Lipschitz, which is a regular condition to ensure Clarke differentiability. However, it may not hold for BiO without LLSC, and we will give the sufficient and necessary conditions for it later in Proposition 4.6. Definition 4.2 adopts the Goldstein stationary points since $\varphi(x)$ can be nonconvex nonsmooth such that traditional stationary points may be intractable, as we will show later in Example 4.1.*

4.1.2 Regularity Conditions for Continuity

Our results underscore that the *sharpness* of LL functions is essential to ensure the tractability of BiO. This is due in part to that the constructions of the hard instances in this section all rely on very *flat* LL functions, as readers will see in §4.2. This observation inspires us to focus on more restricted function classes that possess sharpness to circumvent the ill-conditioned nature of BiO without LLSC. Below, we introduce two conditions that correspond to different degrees of sharpness.

Assumption 4.1 (Gradient Dominance)**.** *Suppose $g(x,y)$ is L-gradient Lipschitz jointly in (x,y), and there exists $\alpha>0$ such that for any $x\in\mathbb{R}^{d_x}, y\in\mathcal{Y}$ we have $\mathcal{G}_{1/L}(y;x)\ge\alpha\,\mathrm{dist}(y,Y^*(x))$.*

Assumption 4.2 (Weak Sharp Minimum)**.** *Suppose $g(x,y)$ is L-Lipschitz in x, and there exists $\alpha>0$ such that for any $x\in\mathbb{R}^{d_x}, y\in\mathcal{Y}$ we have $g(x,y)-g^*(x)\ge 2\alpha\,\mathrm{dist}(y,Y^*(x))$.*

Both conditions are widely used in convex optimization [BF93, DL18]. They are milder conditions than LLSC by allowing $Y^*(x)$ to be non-singleton. Despite being more relaxed, we demonstrate below that either of them can lead to the continuity of $Y^*(x)$ and thus $\varphi(x)$. The continuity of $\varphi(x)$ is crucial for designing algorithms to optimize it.

Proposition 4.2. *Under Assumption 4.1 or 4.2, $Y^*(x)$ is (L/α)-Lipschitz. Furthermore, if $f(x,y)$ is C_f-Lipschitz, then $\varphi(x)$ is $(L/\alpha+1)C_f$-Lipschtz.*

Therefore, the introduced conditions can avoid discontinuous instances such as Example 4.2. It is worth noting that these conditions fundamentally differ from LLSC, as $\varphi(x)$ can be nonsmooth under these conditions, as exemplified below. The potential nonsmoothness of $\varphi(x)$ further justifies the rationality of using Goldstein stationarity in Definition 4.2.

Example 4.1. *Let $f(x,y)=xy$, $g(x,y)=0$ and $\mathcal{Y}=[-1,1]$. We obtain a BiO instance satisfying both Assumptions 4.1 and 4.2. But the resulting $\varphi(x)=-|x|$ is nonsmooth and nonconvex.*

Remark 4.2. *One may wonder how to verify the introduced conditions in applications. It is non-trivial as the value of* $\operatorname{dist}(y, Y^*(x))$ *is unknown. An easy case is Assumption 4.1 with* $\mathcal{Y} = \mathbb{R}^{d_y}$*, which reduces to the Polyak-Lojasiewicz condition [Pol63]:* $\|\nabla_y g(x,y)\|^2 \geq 2\alpha(g(x,y) - g^*(x))$ *by Theorem 2 in [KNS16]. This inequality allows us to identify the following examples that fall into Assumption 4.1:*

Firstly, we can show that Assumption 4.1 strictly covers the LLSC condition.

(i) If g is L-gradient Lipschitz and α-strongly convex, then it satisfies Assumption 4.1.

Secondly, the following example that both AID and ITD fail to optimize satisfies Assumption 4.1.

(ii) Consider the hard BiO instance proposed by [LMY+20]

$$\min_{x \in \mathbb{R}, y \in Y^*(x)} \; (x - y_{[2]})^2 + (y_{[1]} - 1)^2 \qquad Y^*(x) = \arg\min_{y \in \mathbb{R}^2} \; y_{[1]}^2 - 2xy_{[1]}$$

The LL function satisfies Assumption 4.1 with $L = 2$ and $\alpha = 2$.

Thirdly, the BiO with least squares loss studied by [BTTG20] also satisfies Assumption 4.1.[6]

(iii) Consider the BiO with least squares loss

$$\min_{x \in \mathbb{R}^{d_x}, y \in Y^*(x)} \; \frac{1}{2n}\|Ax - y\|^2 \qquad Y^*(x) = \arg\min_{y \in \mathbb{R}^n} \; \frac{1}{2n}\|Ax - y\|_M^2 + \frac{\lambda}{2n}\|y - b\|_M^2$$

where $A \in \mathbb{R}^{n \times d_x}$, $b \in \mathbb{R}^n$ represents the features and labels of the n samples in the dataset, $\lambda > 0$ and M is a positive semi-definite matrix that induces the norm $\|z\|_M = \sqrt{z^\top M z}$. The LL function satisfies Assumption 4.1 with $L = (\lambda + 1)\sigma_{\max}(M)$ and $\alpha = (\lambda + 1)\sigma_{\min}^+(M)$.

4.2 Hardness Results for Intractability

In this subsection, we provide various hardness results to show the challenges of BiO without LLSC. It is a natural idea to tackle BiO without LLSC by adding a regularization term to the LL function and then apply a BiO algorithm designed under LLSC [RFKL19]; however, we will be explaining in the forthcoming Proposition 4.3 that manually regularize the LL function may lead to a huge deviation on the hyper-objective, and thereby does not work as a feasible approach.[7]

Proposition 4.3. *Given a pivot $\widehat{y}$, there exists a BiO instance, where both $f(x,y)$ and $g(x,y)$ are convex in y, and the resulting hyper-objective $\varphi(x)$ is a quadratic function, but for any $\lambda > 0$ the regularized hyper-objective*

$$\varphi_\lambda(x) = \min_{y \in Y_\lambda^*(x)} \; f(x,y) \qquad Y_\lambda^*(x) = \arg\min_{y \in \mathcal{Y}} \; g(x,y) + \lambda\|y - \widehat{y}\|^2$$

is a linear function with $|\inf_{x \in \mathbb{R}^{d_x}} \varphi_\lambda(x) - \inf_{x \in \mathbb{R}^{d_x}} \varphi(x)| = \infty$.

This example indicates that even if the regularization is arbitrarily small, the hyper-objective before and after regularization can be completely different objectives. Consequently, BiO without LLSC should be treated as a distinct research topic from BiO with LLSC.

Till the rest of this section, we demonstrate that both the tasks of finding an LL optimal solution [§4.2.1] and finding a UL stationary point can be intractable for BiO without LLSC [§4.2.2].

[6] We leave more details of this model and its application in adversarial training in §A.3.

[7] The regularization transforms $Y^*(x)$ from a set to a singleton, thus breaking the original problem structure.

4.2.1 Can we Find an LL Optimal Solution?

The goal of finding an LL optimal solution for a given $x \in \mathbb{R}^{d_x}$ is to solve the following problem

$$\min_{y \in Y^*(x)} f(x,y) \qquad Y^*(x) = \arg\min_{y \in \mathcal{Y}} g(x,y) \tag{14}$$

This problem is usually called *simple BiO* [BS14, SS17, KY21] since it involves only one variable y. However, it is not a simple problem as the forthcoming results show its intractability for general convex objectives.

Our lower bound is based on the following first-order zero-chain, which is a generic approach applied extensively in the literature to proving lower bounds for optimization algorithms [NY83, N+18, CDHS20, CDHS21].

Definition 4.4 (Zero-chain)**.** *We call function $h(z) : \mathbb{R}^q \to \mathbb{R}$ a* first-order zero-chain *if for any sequence $\{z_k\}_{k \geq 1}$ satisfying $z_0 = 0$ and*

$$z_i \in \operatorname{Span}\{\partial h(z_0), \ldots, \partial h(z_{i-1})\} \qquad i \geq 1 \tag{15}$$

it holds that $z_{i,[j]} = 0$, $i + 1 \leq j \leq q$.

We remark that in the construction of a zero-chain, we can always assume $z_0 = 0$ without loss of generality. Otherwise, we can translate the function to $h(z - z_0)$. Below, we introduce the convex zero-chain from [N+18, §2.1.2].

Since the subgradients may contain more than one element, we also say $h(z)$ is zero-chain whenever there exists some adversarial subgradient oracle. This would also provide a valid lower bound [N+18].

Definition 4.5 (Gradient Lipschitz Worst-case Zero-chain)**.** *Consider the family of functions*

$$h_q(z) = \frac{1}{8}(z_{[1]} - 1)^2 + \frac{1}{8}\sum_{j=1}^{q-1}(z_{[j+1]} - z_{[j]})^2$$

The following properties hold for any $h_q(z)$ with $q \in \mathbb{N}^+$:

(i) It is a first-order zero-chain

(ii) It has a unique minimizer $z^* = \mathbf{1}$

(iii) It is 1-gradient Lipschitz

In bilevel problems, it is crucial to find a point y that is close to $Y^*(x)$, instead of just achieving a small optimality gap $g(x,y) - g^*(x)$. However, it is difficult for any first-order algorithms to *locate* the minimizers of the function class in Definition 4.5

Proposition 4.4. *Fix an x. For any $K \in \mathbb{N}^+$, there exists $d_y \in \mathbb{N}^+$ such that for any $y_0 \in \mathbb{R}^{d_y}$, there exists a pair of functions $f(x,\cdot), g(x,\cdot)$ that are both convex and 1-gradient Lipschitz, for any first-order algorithm $\mathcal{A}$ which initializes from $y_0 \in \mathcal{Y}$ with $\operatorname{dist}(y_0, y^*(x)) \leq 1$ and generates a sequence of test points $\{y_k\}_{k=0}^K$ with*

$$y_k \in y_0 + \operatorname{Span}\{\nabla_y f(x,y_0), \nabla_y g(x,y_0), \cdots, \nabla_y f(x,y_{k-1}), \nabla_y g(x,y_{k-1})\} \qquad k \geq 1$$

it holds that $|f(x,y_k) - \varphi(x)| \geq 1/4$, where $y^*(x)$ is the unique solution to $\min_{y \in Y^*(x)} f(x,y)$.

The key idea in the proof is to construct the LL function using the worst-case convex zero-chain [N+18], such that any first-order algorithm will require a large number of steps to approach the vicinity of the LL solution mapping $Y^*(x)$.

Next, we prove analogously a lower bound also holds for Lipschitz nonsmooth convex LL functions, using the following function class, which appears in [N+18], §3.2.1.

Definition 4.6 (Lipschitz Zero-chain)**.** *Consider the family of functions*

$$h_q(z) = \frac{\sqrt{q}}{2+\sqrt{q}} \max_{1 \le j \le q} z_{[j]} + \frac{1}{2(2+\sqrt{q})}\|z\|^2$$

The following properties hold for any $h_q(z)$ with $q \in \mathbb{N}^+$:

(i) It is a first-order zero-chain

(ii) It has a unique minimizer $z^* = -\mathbf{1}/\sqrt{q}$

(iii) It is 1-Lipschitz in the unit Euclidean ball $\mathbb{B}(z^*) \triangleq \{z : \|z - z^*\| \le 1\}$

Analogous to Proposition 4.4, we can show the following result.

Proposition 4.5. *Fix an x. For any $K \in \mathbb{N}^+$, there exists $d_y \in \mathbb{N}^+$ such that for any $y_0 \in \mathbb{R}^{d_y}$, there exist there exists a pair of functions $f(x,\cdot), g(x,\cdot)$ that are both convex and 1-Lipschitz on $\mathbb{B}(y^*(x))$, such that for any first-order algorithm $\mathcal{A}$ which initializes from $y_0 \in \mathbb{B}(y^*(x))$, and generates a sequence of test points $\{y_k\}_{k=0}^K$ with*

$$y_k \in y_0 + \mathrm{Span}\{\partial_y f(x,y_0), \partial_y g(x,y_0), \ldots, \partial_y f(x,y_{k-1}), \partial_y g(x,y_{k-1})\} \qquad k \ge 1$$

there exists some subgradients sequence $\{\partial_y f(x,y_0), \partial_y g(x,y_0), \ldots, \partial_y g(x,y_{k-1})\}$ to make $|f(x,y_k) - \varphi(x)| \ge 1/4$ for all k, where $y^*(x)$ is the unique solution to $\min_{y \in Y^*(x)} f(x,y)$.

4.2.2 Can we Find a UL Stationary Point?

Besides the difficulty in finding an LL optimal solution, the goal of finding a UL stationary point is also challenging. Below, we show that the hyper-objective $\varphi(x)$ can be discontinuous without LLSC. Since continuity is one of the basic assumptions for almost all numerical optimization schemes [NW06], our hard instance indicates that $\varphi(x)$ may be intrinsically intractable to optimize for BiO without LLSC.

Example 4.2. *Consider a BiO instance given by*

$$\min_{x \in \mathbb{R}, y \in Y^*(x)} x^2 + y \qquad Y^*(x) = \arg\min_{y \in [-1,1]} -xy$$

It is straightforward to obtain $Y^*(x) = \mathrm{sign}(x)$ and hence the hyper-objective $\varphi(x) = x^2 + \mathrm{sign}(x)$, which is discontinuous at $x = 0$.

In the above example, the discontinuity of $\varphi(x)$ comes from the discontinuity of $Y^*(x) = \mathrm{sign}(x)$. Below, we prove that this statement and its reverse generally holds.

Proposition 4.6. *Suppose the solution mapping $Y^*(x)$ is non-empty and compact for any $x \in \mathbb{R}^{d_x}$.*

(i) If $f(x,y)$ and $Y^*(x)$ are locally Lipschitz, then $\varphi(x)$ is locally Lipschitz

Algorithm 4 SGM($x, y_0, K_0, K, \tau, \theta$)

```
Initialize: $\mathcal{I} = \emptyset$, $\widehat{y}_0 = y_0$
for $k = 0, 1, \ldots, K_0 - 1$ do
    $\widehat{y}_{k+1} = \mathcal{P}_{\mathcal{Y}}\left[\widehat{y}_k - \tau \partial_y g(x, \widehat{y}_k)\right]$
end for
$\widehat{g}^*(x) = g(x, \widehat{y}_{K_0})$
for $k = 0, 1, \ldots, K - 1$ do
    if $g(x, y_k) - \widehat{g}^*(x) \leq 2\theta$ then
        $y_{k+1} = \mathcal{P}_{\mathcal{Y}}\left[y_k - \tau \partial_y f(x, y_k)\right]$
        $\mathcal{I} = \mathcal{I} \cup \{k\}$
    else
        $y_{k+1} = \mathcal{P}_{\mathcal{Y}}\left[y_k - \tau \partial_y g(x, y_k)\right]$
    end if
end for
$y_{\mathrm{out}} = \frac{1}{|\mathcal{I}|} \sum_{k \in \mathcal{I}} y_k$
Output: $y_{\mathrm{out}}$
```

(ii) Conversely, if $\varphi(x)$ is locally Lipschitz for any locally Lipschitz function $f(x, y)$, then $Y^*(x)$ is locally Lipschitz

(iii) If $f(x, y)$ is C_f-Lipschitz and $Y^*(x)$ is κ-Lipschitz, then $\varphi(x)$ is C_φ-Lipschitz with coefficient $C_\varphi = (\kappa + 1)C_f$

(iv) Conversely, if $\varphi(x)$ is C_φ-Lipschitz for any C_f-Lipschitz function $f(x, y)$, then $Y^*(x)$ is κ-Lipschitz with coefficient $\kappa = C_\varphi / C_f$

Local Lipschitz continuity ensures UL stationary points (Definition 4.2) are well-defined, while global Lipschitz continuity enables uniform complexity bounds for non-asymptotic analysis (as we will use in §4.3.2). According to the above theorem, ensuring the continuity of $Y^*(x)$ is the key to obtaining the desired continuity of $\varphi(x)$. This motivates us to focus on well-behaved LL functions that confer continuity of $Y^*(x)$.

4.3 The Proposed Methods

In this subsection, we propose novel polynomial time algorithms for BiO under Assumption 4.1 and 4.2. We first borrow ideas from switching gradient methods to overcome the difficulty of multiple LL minima [§4.3.1], and then propose a method motivated by gradient-free optimization that can provably converge to a UL stationary point [§4.3.2].

4.3.1 Finding LL Optimality via Switching Gradient Method

In (14), the LL constraint $y \in Y^*(x)$ is equivalent to an inequality constraint $g(x, y) \leq g^*(x)$. Based on this observation, we generalize *Polyak's Switching Gradient Method* [Pol67] for the functional constrained problems to Algorithm 4 when the following assumptions hold.

Assumption 4.3. *Suppose that*

(i) both $f(x, y)$ and $g(x, y)$ are convex in y

Algorithm 5 $\texttt{IGFM}(x_0, y_0, \eta, T, \delta, K_0, K, \tau, \theta)$

```
Input: Sub-routine $\mathcal{A}$ can estimate $\widetilde{\varphi}(x) \approx \varphi(x)$ for any $x \in \mathbb{R}^{d_x}$
for $t = 0, 1, \dots, T-1$ do
    Sample $u_t \in \mathbb{R}^{d_x}$ uniformly from the unit sphere $\partial\mathbb{B}_1$ in $\mathbb{R}^{d_x}$
    Estimate $\widetilde{\varphi}(x_t + \delta u_t)$ and $\widetilde{\varphi}(x_t - \delta u_t)$ by sub-routine $\mathcal{A}$
    $\widehat{\nabla}_t = \frac{d_x}{2\delta}\left(\widetilde{\varphi}(x_t + \delta u_t) - \widetilde{\varphi}(x_t - \delta u_t)\right) u_t$
    $x_{t+1} = x_t - \eta \widehat{\nabla}_t$
end for
Output: $x_{\text{out}}$ uniformly chosen from $\{x_t\}_{t=0}^{T-1}$
```

(ii) *$\mathcal{Y}$ is compact with diameter R*

(iii) *$f(x, y)$ is C_f-Lipschitz on $\mathbb{R}^{d_x} \times \mathcal{Y}$*

(iv) *$g(x, \cdot)$ is C_g-Lipschitz on $\mathcal{Y}$ for any $x \in \mathbb{R}^{d_x}$*

(v) *either Assumption 4.1 or 4.2 holds for $g(x, y)$*

Under the above assumptions, we can prove the following result.

Theorem 4.1. *Fix an x. Under Assumption 4.3, Algorithm 4 with appropriate parameters can ouput a point y_{out} satisfying $|f(x, y_{\text{out}}) - \varphi(x)| \leq \zeta$ and $g(x, y_{\text{out}}) - g^*(x) \leq \zeta$ with $\mathcal{O}(\text{poly}(1/\zeta))$ first-order oracle calls from g.*

The corresponding proof and specific parameters of the algorithm can be found in §4.4 and §C.

4.3.2 Finding UL Stationarity via Gradient-Free Method

Without LLSC, the hyper-gradient $\nabla\varphi(x)$ may not have an explicit form as (4). To tackle this challenge, we propose the *Inexact Gradient-Free Method* (`IGFM`) in Algorithm 5. The algorithm is motivated by recent advances in nonsmooth nonconvex gradient-free optimization [LZJ22]. Our (zeroth-order) oracle query $\widetilde{\varphi}(x) \approx \varphi(x)$ is *inexact* since it is an approximation from a sub-routine $\mathcal{A}$. Below, we show that when $\mathcal{A}$ can guarantee sufficient approximation precision, `IGFM` provably finds a Goldstein stationary point of a Lipschitz hyper-objective function $\varphi(x)$.

Assumption 4.4. *Suppose that*

(i) *$\varphi(x)$ is C_φ-Lipschitz*

(ii) *$\mathcal{A}$ ensures $|\widetilde{\varphi}(x) - \varphi(x)| \leq \mathcal{O}(\delta\varepsilon^2/(d_x C_\varphi))$ for any $x \in \mathbb{R}^{d_x}$*

Theorem 4.2. *Given any $\varepsilon \lesssim C_f$. Suppose the hyper-objective $\varphi(x) = \min_{y \in Y^*(x)} f(x, y)$ has a finite minimum value denoted by $\varphi^* = \inf_{x \in \mathbb{R}^{d_x}} \varphi(x) > -\infty$, and let $\Delta = \varphi(x_0) - \varphi^*$. Under Assumption 4.4, set*

$$T = \mathcal{O}\left(d_x^{3/2}\left(\frac{C_\varphi^4}{\varepsilon^4} + \frac{\Delta C_\varphi^3}{\delta\varepsilon^4}\right)\right) \qquad \eta = \Theta\left(\sqrt{\frac{\delta(\Delta + \delta C_\varphi)}{d_x^{3/2} C_\varphi^3 T}}\right) \tag{16}$$

Then Algorithm 5 can output a point x_{out} that satisfies $\mathbb{E}\min\{\|s\| : s \in \partial_\delta\varphi(x_{\text{out}})\} \leq \varepsilon$.

Now it remains to verify Assumption 4.4. Note Assumption 4.4(i) can be verified by Proposition 4.6, while Assumption 4.4(ii) can be verified by Theorem 4.1. Therefore we have the following result. To the best of our knowledge, it is among the first theoretical analysis that shows the non-asymptotic convergence to a UL stationary point for BiO without LLSC:

Corollary 4.1. *Suppose Assumption 4.3 holds. Set $\mathcal{A}$ as the* `SGM` *Algorithm 4. Then Algorithm 5 with appropriate parameters can output a (δ, ϵ)-Goldstein stationary point of $\varphi(x)$ in expectation within $\mathcal{O}(\mathrm{poly}(d_x, 1/\varepsilon, 1/\delta))$ zeroth-order and first-order oracle calls from f and g.*

4.4 Proof of Theorem 4.1

To prove Theorem 4.1 we first prove that the proposed Switching (sub)Gradient Method in Algorithm 4 can find an LL optimal solution under the following Hölderian error bound condition.

Assumption 4.5. *We suppose the LL function $g(x, \cdot)$ satisfies the r-*th order Hölderian error bound condition *on set $\mathcal{Y}$ with some coefficient $\nu > 0$, that is*

$$\frac{\nu}{r}\,\mathrm{dist}(y, Y^*(x))^r \le g(x,y) - g^*(x) \qquad \forall y \in \mathcal{Y}$$

Note that this condition is also used by [JAMH23] and they show the following result

Lemma 4.1 (Proposition 1 in [JAMH23])**.** *Suppose that Assumption 4.5 holds, $f(x, \cdot)$ is convex and $f(x,y)$ is C_f-Lipschitz. If a point y satisfies*

$$f(x,y) - \varphi(x) \le \zeta \qquad g(x,y) - g^*(x) \le \frac{\nu}{r}\left(\frac{\zeta}{C_f}\right)^r \tag{17}$$

then we have $|f(x,y) - \varphi(x)| \le \zeta$.

(17) can be achieved by the `SGM`, then we can show the following result for finding an LL optimal solution the Hölderian error bound condition.

Theorem 4.3. *Under Assumptions 4.3 and 4.5 we let*

$$\theta = \min\left\{\zeta, \frac{\nu}{4r}\left(\frac{\zeta}{C_f}\right)^r\right\} \qquad K_0 = K = \left\lceil \frac{4R^2 \max\{C_f^2, C_g^2\}}{\theta^2} \right\rceil \qquad \tau = \frac{R}{\max\{C_f, C_g\}\sqrt{K}} \tag{18}$$

then Algorithm 4 can output a point y_{out} satisfying $|f(x, y_{\mathrm{out}}) - \varphi(x)| \le \zeta$ within $\mathcal{O}\left(\frac{r^2 \max\{C_f^2, C_g^2\} C_f^{2r} R^2}{\nu^2 \zeta^{2r}}\right)$ first-order oracle complexity.

We introduce next Lemma 4.3 which relies on the following standard lemma for subgradient descent.

Lemma 4.2 (Subgradient Descent)**.** *Suppose h is a L-Lipschitz convex function. For any $y, z \in \mathcal{Y}$, if we let $y^+ = \mathcal{P}_{\mathcal{Y}}\left[y - \tau \partial h(y)\right]$, then it holds that*

$$h(y) - h(z) \le \frac{1}{2\tau}\left(\|y - z\|^2 - \|y^+ - z\|^2\right) + \frac{\tau L^2}{2}$$

Proof of Lemma 4.2. See Theorem 3.2 in [B+15]. □

Using this lemma, we then show the following result.

Lemma 4.3. *Under the setting of Theorem 4.3, the output of Algorithm 4 satisfies*

$$f(x, y_{\mathrm{out}}) - \varphi(x) \le \theta \qquad g(x, y_{\mathrm{out}}) - g^*(x) \le 4\theta$$

Hence, Theorem 4.3 follows naturally by combining Lemma 4.1 and Lemma 4.3.

Proof of Lemma 4.3. By Theorem 3.2 in [B+15], the initialization step ensures $\widehat{g}^*(x) - g^*(x) \le 2\theta$. Pick any $y^*(x) \in \arg\min_{y \in Y^*(x)} f(x, y)$ and denote $C = \max\{C_f, C_g\}$. According to Lemma 4.2 we obtain

$$f(x, y_k) - \varphi(x) \le \frac{1}{2\tau}\left(\|y_k - y^*(x)\|^2 - \|y_{k+1} - y^*(x)\|^2\right) + \frac{\tau C^2}{2} \qquad k \in \mathcal{I}$$
$$g(x, y_k) - g^*(x) \le \frac{1}{2\tau}\left(\|y_k - y^*(x)\|^2 - \|y_{k+1} - y^*(x)\|^2\right) + \frac{\tau C^2}{2} \qquad k \in \mathcal{I}^c$$

Combing them together yields

$$\frac{1}{K}\sum_{k \in \mathcal{I}} f(x, y_k) - \varphi(x) + \frac{1}{K}\sum_{k \in \mathcal{I}^c} g(x, y_k) - g^*(x) \le \frac{R^2}{2\tau K} + \frac{\tau C^2}{2} = \frac{RC}{\sqrt{K}} \tag{19}$$

With (19) in hand, it suffices to show the result. Firstly, we show that $\mathcal{I} \ne \emptyset$, and thus y_{out} is well-defined. Otherwise, we would have the following contradiction

$$2\theta \le \frac{1}{K}\sum_{k=0}^{K-1} g(x, y_k) - \widehat{g}^*(x) \le \frac{1}{K}\sum_{k=0}^{K-1} g(x, y_k) - g^*(x) \le \frac{RC}{\sqrt{K}} \le \frac{\theta}{2}$$

Secondly, we show that the output will not violate the constraint too much by

$$g(x, y_{\mathrm{out}}) - g^*(x) \le \frac{1}{|\mathcal{I}|}\sum_{k \in \mathcal{I}} (g(x, y_k) - \widehat{g}^*(x)) + (\widehat{g}^*(x) - g^*(x)) \le 4\theta$$

Thirdly, we show that $f(x, y_{\mathrm{out}}) - \varphi(x) \le \theta$. It is trivial when $\sum_{k \in \mathcal{I}} f(x, y_k) - \varphi(x) \le 0$ since it is an immediate result of Jensen's inequality. Therefore we can only focus on the case when $\sum_{k \in \mathcal{I}} f(x, y_k) - \varphi(x) > 0$. In this case, we can show that $|\mathcal{I}| \ge K/2$, otherwise we would have

$$\theta < \frac{1}{K}\sum_{k \in \mathcal{I}^c} g(x, y_k) - \widehat{g}^*(x) \le \frac{1}{K}\sum_{k \in \mathcal{I}^c} g(x, y_k) - g^*(x) \le \frac{RC}{\sqrt{K}} \le \frac{\theta}{2}$$

which also leads to a contradiction. Hence we must have $|\mathcal{I}| \ge K/2$, therefore, we obtain

$$f(x, y_{\mathrm{out}}) - \varphi(x) \le \frac{1}{|\mathcal{I}|}\sum_{k \in \mathcal{I}} f(x, y_k) - \varphi(x) \le \frac{2}{K}\sum_{k \in \mathcal{I}} f(x, y_k) - \varphi(x) \le \frac{2RC}{\sqrt{K}} \le \theta$$

This completes our proof. □

We want to use Theorem 4.3 to prove Theorem 4.1. The only difference between them relies upon the assumption. The following proposition shows that both Assumptions 4.1 and 4.2 imply Assumption 4.5 when $g(x, y)$ is convex in y. Therefore, the function class studied in Theorem 4.1 is contained in the function class studied in Theorem 4.3.

Proposition 4.7. *If $g(x, \cdot)$ is convex, then either Assumption 4.1 or 4.2 implies Assumption 4.5.*

Proof of Proposition 4.7. According to Corollary 3.6 in [DL18], Assumption 4.1 implies Assumption 4.5 with any $\nu < \alpha$ under the convexity of $g(x, \cdot)$. For Assumption 4.2, it is clear that it is equivalent to Assumption 4.5 with $r = 1$. □

Theorem 4.1 naturally follows by combining Theorem 4.3 and Proposition 4.7.

4.5 Proof of Theorem 4.2

In order to show Theorem 4.2, we let $\varphi_\delta \triangleq \mathbb{E}_{v\sim\mathbb{P}_v}[\varphi(x + \delta v)]$ where $\mathbb{P}_v$ is a uniform distribution on a unit ball in ℓ_2-norm. Then, we define

$$\nabla_t \triangleq \frac{d_x}{2\delta}\left(\varphi(x_t + \delta u_t) - \varphi(x_t - \delta u_t)\right) u_t \tag{20}$$

According to Lemma D.1 in [LZJ22], ∇_t satisfies the following properties

$$\mathbb{E}_{u_t}\left[\nabla_t \mid x_t\right] = \nabla\varphi_\delta(x_t) \qquad \mathbb{E}_{u_t}[\|\nabla_t\|^2 \mid x_t] \leq 16\sqrt{2\pi} d_x C_\varphi^2$$

Then we know that

$$\mathbb{E}_{u_t}[\|\nabla_t - \widehat{\nabla}_t\| \mid x_t] \leq \frac{d_x\zeta}{\delta}\mathbb{E}_{u_t}\|u_t\| = \frac{d_x\zeta}{\delta} \leq \frac{c_4\varepsilon^2}{C_\varphi} \tag{21}$$

and

$$\begin{aligned}
&\mathbb{E}_{u_t}[\|\widehat{\nabla}_t\|^2 \mid x_t] \leq 2\mathbb{E}_{u_t}[\|\nabla_t\|^2 \mid x_t] + 2\mathbb{E}_{u_t}[\|\nabla_t - \widehat{\nabla}_t\|^2 \mid x_t] \\
&\leq 2\mathbb{E}_{u_t}[\|\nabla_t\|^2 \mid x_t] + \frac{2d_x^2\zeta^2}{\delta^2}\mathbb{E}_{u_t}\|u_t\|^2 \leq 32\sqrt{2\pi} d_x C_\varphi^2 + \frac{2d_x^2\zeta^2}{\delta^2} \leq c_1 d_x C_\varphi^2
\end{aligned} \tag{22}$$

for some positive constant $c_1, c_4 > 0$. Then we use the results of (21), (22) as well as the standard analysis of gradient descent to obtain

$$\begin{aligned}
&\mathbb{E}\left[\varphi_\delta(x_{t+1}) \mid x_t\right] \leq \varphi_\delta(x_t) - \eta\left\langle\nabla\varphi_\delta(x_t), \mathbb{E}[\widehat{\nabla}_t \mid x_t]\right\rangle + \frac{c_2\eta^2 C_\varphi\sqrt{d_x}}{2\delta}\mathbb{E}[\|\widehat{\nabla}_t\|^2 \mid x_t] \\
&\leq \varphi_\delta(x_t) - \eta\|\nabla\varphi_\delta(x_t)\|^2 + \frac{\eta C_\varphi d_x\zeta}{\delta} + \frac{c_2\eta^2 C_\varphi\sqrt{d_x}}{2\delta}\mathbb{E}[\|\widehat{\nabla}_t\|^2 \mid x_t] \\
&\leq \varphi_\delta(x_t) - \eta\left\|\nabla\varphi_\delta(x_t)\right\|^2 + \frac{c_3\eta^2 C_\varphi^3 d_x^{3/2}}{\delta} + \eta c_4\varepsilon^2
\end{aligned}$$

where we use Proposition 2.3 in [LZJ22] that φ_δ is differentiable and C_φ-Lipschitz with the $(c_2 C_\varphi\sqrt{d_x}/\delta)$-Lipschitz gradient where $c_2 > 0$ is a positive constant and we define $c_3 = 2c_1c_2$. Telescoping for $t = 0, 1, \ldots, T$, we obtain

$$\mathbb{E}\left\|\nabla\varphi_\delta(x_{\text{out}})\right\|^2 \leq \frac{\Delta + \delta C_\varphi}{\eta T} + \frac{c_3\eta C_\varphi^3 d^{3/2}}{\delta} + c_4\varepsilon^2$$

where we use $|\varphi_\delta(x) - \varphi(x)| \leq \delta C_\varphi$ for any $x \in \mathbb{R}^{d_x}$ by Proposition 2.3 in [LZJ22].

Lastly, plugging the value of η, T with a sufficiently small constant c_4 and noting that $\nabla\varphi(x_{\text{out}}) \in \partial_\delta\varphi(x_{\text{out}})$ by Theorem 3.1 in [LZJ22], we arrive at the conclusion.

5 Conclusion

In significance, our proposed algorithms in optimizing bilevel optimization (BiO) problems with or without LLSC is underscored by its state-of-the-art convergence rates and computational efficiency.

For BiO with LLSC, we have presented the `(P)RAF`^{2}`BA` algorithm that leverages *fully* first-order oracles to find approximate stationary points in nonconvex-strongly-convex BiO, enhancing oracle complexity for efficient optimization. Theoretical guarantees for finding approximate first-order stationary points and second-order stationary points with state-of-the-art query complexities have been established, showcasing their effectiveness in solving complex optimization tasks. In particular when applied to minimax optimization problem, we recovered `PRAGDA` that achieves the state-of-the-art in finding approximate second-order stationary point of the hyper-objective objective.

For BiO without LLSC, we first identified several regularity conditions of the LL problems that can provably confer tractability. Then we presented hardness results showing that BiO for general convex LL functions is intractable to solve. Finally we proposed `IGFM`, which uses `SGM` as an efficient sub-routine to find an approximate stationary point of the hyper-objective in polynomial time.

Although this paper focuses primarily on the theoretical level, we expect our results can shed light on efficient algorithm design for BiO applications in practice. We also hope our work can be a good starting point for non-asymptotic analysis for more challenging BiO problems, such as BiO with nonconvex LL functions or BiO with intertwined inequality constraints $h(x, y) \leq 0$.

References

[AAZB$^+$17] Naman Agarwal, Zeyuan Allen-Zhu, Brian Bullins, Elad Hazan, and Tengyu Ma. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1195–1199, 2017.

[ABGLP19] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[ACB17] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223. PMLR, 2017.

[AM22a] Michael Arbel and Julien Mairal. Amortized implicit differentiation for stochastic bilevel optimization. In *The Tenth International Conference on Learning Representations*, 2022.

[AM22b] Michael Arbel and Julien Mairal. Non-convex bilevel games with critical point selection maps. *Advances in Neural Information Processing Systems*, 35:8013–8026, 2022.

[B$^+$15] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[BF93] James V Burke and Michael C Ferris. Weak sharp minima in mathematical programming. *SIAM Journal on Control and Optimization*, 31(5):1340–1359, 1993.

[BHTV19] Luca Bertinetto, Joao F Henriques, Philip Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. In *International Conference on Learning Representations*, 2019.

[BLPSF21] Jérôme Bolte, Tam Le, Edouard Pauwels, and Tony Silveti-Falls. Nonsmooth implicit differentiation for machine-learning and optimization. *Advances in neural information processing systems*, 34:13537–13549, 2021.

[BS11] Michael Brückner and Tobias Scheffer. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 547–555, 2011.

[BS14] Amir Beck and Shoham Sabach. A first order method for finding minimal norm-like solutions of convex optimization problems. *Mathematical Programming*, 147(1):25–46, 2014.

[BTTG20] Nicholas Bishop, Long Tran-Thanh, and Enrico Gerding. Optimal learning from verified training data. *Advances in Neural Information Processing Systems*, 33:9520–9529, 2020.

[CDHS17] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. "convex until proven guilty": dimension-free acceleration of gradient descent on non-convex functions. In *International conference on machine learning*, pages 654–663. PMLR, 2017.

[CDHS18] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Accelerated methods for nonconvex optimization. *SIAM Journal on Optimization*, 28(2):1751–1772, 2018.

[CDHS20] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points I. *Mathematical Programming*, 184(1):71–120, 2020.

[CDHS21] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points II: first-order methods. *Mathematical Programming*, 185(1):315–355, 2021.

[CHL$^+$23] Ziyi Chen, Zhengyang Hu, Qunwei Li, Zhe Wang, and Yi Zhou. A cubic regularization approach for finding local minimax points in nonconvex minimax optimization. *Transactions on Machine Learning Research*, 2023.

[CL11] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

[Cla90] Frank H Clarke. *Optimization and nonsmooth analysis*. SIAM, 1990.

[Cla17] Christian Clason. Nonsmooth analysis and optimization. *arXiv preprint arXiv:1708.04180*, 2017.

[CMO23] Ashok Cutkosky, Harsh Mehta, and Francesco Orabona. Optimal stochastic non-smooth non-convex optimization through online-to-non-convex conversion. In *International Conference on Machine Learning*, pages 6643–6670. PMLR, 2023.

[CMZ23] Lesi Chen, Yaohua Ma, and Jingzhao Zhang. Near-optimal fully first-order algorithms for finding stationary points in bilevel optimization. *arXiv preprint arXiv:2306.14853*, 2023.

[CRS17] Frank E Curtis, Daniel P Robinson, and Mohammadreza Samadi. A trust region algorithm with a worst-case iteration complexity of $\mathcal{O}(\epsilon^{-3/2})$ for nonconvex optimization. *Mathematical Programming*, 162:1–32, 2017.

[CSXY22] Tianyi Chen, Yuejiao Sun, Quan Xiao, and Wotao Yin. A single-timescale method for stochastic bilevel optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 2466–2488. PMLR, 2022.

[CSY21] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. *Advances in Neural Information Processing Systems*, 34:25294–25307, 2021.

[Dan12] John M Danskin. *The theory of max-min, with applications*, volume 5. Springer Science & Business Media, 2012.

[DAVM22] Mathieu Dagréou, Pierre Ablin, Samuel Vaiter, and Thomas Moreau. A framework for bilevel optimization that enables stochastic and global variance reduction algorithms. *Advances in Neural Information Processing Systems*, 35:26698–26710, 2022.

[DDG$^+$22] Marina Danilova, Pavel Dvurechensky, Alexander Gasnikov, Eduard Gorbunov, Sergey Guminov, Dmitry Kamzolov, and Innokentiy Shibaev. Recent theoretical advances in non-convex optimization. In *High-Dimensional Optimization and Probability: With a View Towards Data Science*, pages 79–163. Springer, 2022.

[DDL$^+$22] Damek Davis, Dmitriy Drusvyatskiy, Yin Tat Lee, Swati Padmanabhan, and Guanghao Ye. A gradient sampling method with complexity guarantees for Lipschitz functions in high and low dimensions. *Advances in neural information processing systems*, 35:6692–6703, 2022.

[Dem02] Stephan Dempe. *Foundations of Bilevel Programming.* Springer Science & Business Media, 2002.

[DKK06] Stephan Dempe, Vyatcheslav V Kalashnikov, and Nataliya Kalashnykova. Optimality conditions for bilevel programming problems. *Optimization with Multivalued Mappings: Theory, Applications, and Algorithms*, pages 3–28, 2006.

[DL18] 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.

[Dom12] Justin Domke. Generic methods for optimization-based modeling. In *International Conference on Artificial Intelligence and Statistics*, volume 22, pages 318–326. PMLR, 2012.

[DSAP22] Robert Dyro, Edward Schmerling, Nikos Arechiga, and Marco Pavone. Second-order sensitivity analysis for bilevel optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 9166–9181. PMLR, 2022.

[DZ20] Stephan Dempe and Alain Zemkoho. *Bilevel Optimization: Advances and Next Challenges*, volume 161. Springer, 2020.

[FDFP17] Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pages 1165–1173. PMLR, 2017.

[FFS$^+$18] 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.

[FH19] Matthias Feurer and Frank Hutter. Hyperparameter optimization. In *Automated Machine Learning: Methods, Systems, Challenges*, pages 3–33. Springer International Publishing, 2019.

[GFC$^+$16] Stephen Gould, Basura Fernando, Anoop Cherian, Peter Anderson, Rodrigo Santa Cruz, and Edison Guo. On differentiating parameterized argmin and argmax problems with application to bi-level optimization. *arXiv preprint arXiv:1607.05447*, 2016.

[GFPS20] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning*, pages 3748–3758. PMLR, 2020.

[GHZY21] Zhishuai Guo, Quanqi Hu, Lijun Zhang, and Tianbao Yang. Randomized stochastic variance-reduced methods for multi-task stochastic bilevel optimization. *arXiv preprint arXiv:2105.02266*, 2021.

[Gol77] A.A. Goldstein. Optimization of Lipschitz continuous functions. *Mathematical Programming*, 13:14–22, 1977.

[GPAM$^+$20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[GSS14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[GW18] Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

[HAMS21] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169, 2021.

[HJML22] Minhui Huang, Kaiyi Ji, Shiqian Ma, and Lifeng Lai. Efficiently escaping saddle points in bilevel optimization. *arXiv preprint arXiv:2202.03684*, 2022.

[HML21] Minhui Huang, Shiqian Ma, and Lifeng Lai. A Riemannian block coordinate descent method for computing the projection robust Wasserstein distance. In *International Conference on Machine Learning*, pages 4446–4455. PMLR, 2021.

[HWWY23] Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1):147–180, 2023.

[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*, pages 10305–10323. PMLR, 2023.

[JGN$^+$17] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1724–1732, 2017.

[JKL$^+$23] Michael Jordan, Guy Kornowski, Tianyi Lin, Ohad Shamir, and Manolis Zampetakis. Deterministic nonsmooth nonconvex optimization. In *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195, pages 4570–4597. PMLR, 2023.

[JL23] Kaiyi Ji and Yingbin Liang. Lower bounds and accelerated algorithms for bilevel optimization. *Journal of Machine Learning Research*, 24(22):1–56, 2023.

[JLLP20] Kaiyi Ji, Jason D Lee, Yingbin Liang, and H Vincent Poor. Convergence of meta-learning with task-specific adaptation over partial parameters. *Advances in Neural Information Processing Systems*, 33:11490–11500, 2020.

[JLLY22] Kaiyi Ji, Mingrui Liu, Yingbin Liang, and Lei Ying. Will bilevel optimizers benefit from loops. *Advances in Neural Information Processing Systems*, 35:3011–3023, 2022.

[JNJ18] Chi Jin, Praneeth Netrapalli, and Michael I Jordan. Accelerated gradient descent escapes saddle points faster than gradient descent. In *Conference On Learning Theory*, pages 1042–1085. PMLR, 2018.

[JNJ20] Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International conference on machine learning*, pages 4880–4889. PMLR, 2020.

[JV22] Yibo Jiang and Victor Veitch. Invariant and transportable representations for anti-causal domain shifts. *Advances in Neural Information Processing Systems*, 35:20782–20794, 2022.

[JYL21] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*, pages 4882–4892. PMLR, 2021.

[KKWN23] Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method for stochastic bilevel optimization. In *International Conference on Machine Learning*, pages 18083–18113. PMLR, 2023.

[KNS16] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pages 795–811. Springer, 2016.

[KS21] Guy Kornowski and Ohad Shamir. Oracle complexity in nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems*, 34:324–334, 2021.

[KS24] Guy Kornowski and Ohad Shamir. An algorithm with optimal dimension-dependence for zero-order nonsmooth nonconvex stochastic optimization. *Journal of Machine Learning Research*, 25(122):1–14, 2024.

[KT99] Vijay Konda and John Tsitsiklis. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

[KY21] 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.

[KZH+21] Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. *Advances in neural information processing systems*, 34:30271–30283, 2021.

[LBBH98] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[LFH+20] Tianyi Lin, Chenyou Fan, Nhat Ho, Marco Cuturi, and Michael Jordan. Projection robust Wasserstein distance and Riemannian optimization. *Advances in neural information processing systems*, 33:9383–9397, 2020.

[LHH22] Junyi Li, Feihu Huang, and Heng Huang. Local stochastic bilevel optimization with momentum-based variance reduction. *arXiv preprint arXiv:2205.01608*, 2022.

[LJJ20a] Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pages 6083–6093. PMLR, 2020.

[LJJ20b] Tianyi Lin, Chi Jin, and Michael I Jordan. Near-optimal algorithms for minimax optimization. In *Conference on Learning Theory*, pages 2738–2779. PMLR, 2020.

[LL23] Huan Li and Zhouchen Lin. Restarted nonconvex accelerated gradient descent: No more polylogarithmic factor in the $O(\epsilon^{-7/4})$ complexity. *Journal of Machine Learning Research*, 24(157):1–37, 2023.

[LLC22] Luo Luo, Yujun Li, and Cheng Chen. Finding second-order stationary points in nonconvex-strongly-concave minimax optimization. *Advances in Neural Information Processing Systems*, 35:36667–36679, 2022.

[LLY+21] Risheng Liu, Xuan Liu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A value-function-based interior-point method for non-convex bi-level optimization. In *International conference on machine learning*, pages 6882–6892. PMLR, 2021.

[LLZZ21] Risheng Liu, Yaohua Liu, Shangzhi Zeng, and Jin Zhang. Towards gradient-based bilevel optimization with non-convex followers and beyond. *Advances in Neural Information Processing Systems*, 34:8662–8675, 2021.

[LM23] Zhaosong Lu and Sanyou Mei. A first-order augmented Lagrangian method for constrained minimax optimization. *arXiv preprint arXiv:2301.02060*, 2023.

[LM24] Zhaosong Lu and Sanyou Mei. First-order penalty methods for bilevel optimization. *SIAM Journal on Optimization*, 34(2):1937–1969, 2024.

[LMY+20] Risheng Liu, Pan Mu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *International conference on machine learning*, pages 6305–6315. PMLR, 2020.

[LSR$^+$22] Lars Lorch, Scott Sussex, Jonas Rothfuss, Andreas Krause, and Bernhard Schölkopf. Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35:13104–13118, 2022.

[LSY19] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *International Conference on Learning Representations*, 2019.

[LTHC20] Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications. *IEEE Transactions on Signal Processing*, 68:3676–3691, 2020.

[LYHZ20] Luo Luo, Haishan Ye, Zhichao Huang, and Tong Zhang. Stochastic recursive gradient descent ascent for stochastic nonconvex-strongly-concave minimax problems. *Advances in Neural Information Processing Systems*, 33:20566–20577, 2020.

[LYW$^+$22] Bo Liu, Mao Ye, Stephen Wright, Peter Stone, and Qiang Liu. BOME! bilevel optimization made easy: A simple first-order approach. *Advances in neural information processing systems*, 35:17248–17262, 2022.

[LZJ22] Tianyi Lin, Zeyu Zheng, and Michael Jordan. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems*, 35:26160–26175, 2022.

[MDA15] Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pages 2113–2122. PMLR, 2015.

[N$^+$18] Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

[NP06] Yurii Nesterov and Boris T Polyak. Cubic regularization of Newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[NSH$^+$19] Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *Advances in Neural Information Processing Systems*, 32, 2019.

[NW06] Jorge Nocedal and Stephen J Wright. *Numerical Optimization*. Springer-Verlag, New York, 2nd edition, 2006.

[NY83] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[Ped16] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*, pages 737–746. PMLR, 2016.

[PLSS21] Quang Pham, Chenghao Liu, Doyen Sahoo, and HOI Steven. Contextual transformation networks for online continual learning. In *International Conference on Learning Representations*, 2021.

[Pol63] Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 3(4):643–653, 1963.

[Pol67] Boris Teodorovich Polyak. A general method for solving extremal problems. In *Doklady Akademii Nauk*, volume 174, pages 33–36. Russian Academy of Sciences, 1967.

[Rd17] Vincent Roulet and Alexandre d'Aspremont. Sharpness, restart and acceleration. *Advances in Neural Information Processing Systems*, 30, 2017.

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

[RL17] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International conference on learning representations*, 2017.

[RW09] R Tyrrell Rockafellar and Roger J-B Wets. *Variational Analysis*, volume 317. Springer Science & Business Media, 2009.

[SC23] Han Shen and Tianyi Chen. On penalty-based bilevel gradient descent method. In *International Conference on Machine Learning*, pages 30992–31015. PMLR, 2023.

[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*, pages 1723–1732. PMLR, 2019.

[SJGL22] Daouda Sow, Kaiyi Ji, Ziwei Guan, and Yingbin Liang. A primal-dual approach to bilevel optimization with multiple inner minima. *arXiv preprint arXiv:2203.01123*, 2022.

[SND18] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018.

[SS17] Shoham Sabach and Shimrit Shtern. A first order method for solving convex bilevel optimization problems. *SIAM Journal on Optimization*, 27(2):640–660, 2017.

[SYL$^+$23] Zhenqian Shen, Hansi Yang, Yong Li, James Kwok, and Quanming Yao. Efficient hyper-parameter optimization with cubic regularization. *Advances in Neural Information Processing Systems*, 36, 2023.

[SZB20] Bradly Stadie, Lunjun Zhang, and Jimmy Ba. Learning intrinsic rewards as a bi-level optimization problem. In *Conference on Uncertainty in Artificial Intelligence*, pages 111–120. PMLR, 2020.

[TSJ$^+$18] Nilesh Tripuraneni, Mitchell Stern, Chi Jin, Jeffrey Regier, and Michael I Jordan. Stochastic cubic regularization for fast nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

[TZS22] Lai Tian, Kaiwen Zhou, and Anthony Man-Cho So. On the finite-time complexity and practical computation of approximate stationarity concepts of Lipschitz functions. In *International Conference on Machine Learning*, pages 21360–21379. PMLR, 2022.

[WCJ$^+$21] 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.

[WGS$^+$22] Xiaoxing Wang, Wenxuan Guo, Jianlin Su, Xiaokang Yang, and Junchi Yan. ZARTS: On zero-order optimization for neural architecture search. *Advances in Neural Information Processing Systems*, 35:12868–12880, 2022.

[WHJ$^+$22] 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.

[WL20] Yuanhao Wang and Jian Li. Improved algorithms for convex-concave minimax optimization. *Advances in Neural Information Processing Systems*, 33:4800–4810, 2020.

[XLC23] Quan Xiao, Songtao Lu, and Tianyi Chen. An alternating optimization method for bilevel problems under the Polyak-Łojasiewicz condition. *Advances in Neural Information Processing Systems*, 36, 2023.

[YJL21] Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. *Advances in Neural Information Processing Systems*, 34:13670–13682, 2021.

[YLL$^+$23] Haikuo Yang, Luo Luo, Chris Junchi Li, Michael Jordan, and Maryam Fazel. Accelerating inexact hypergradient descent for bilevel optimization. In *OPT 2023: Optimization for Machine Learning*, 2023.

[YZ95] Jane J Ye and DL Zhu. Optimality conditions for bilevel programming problems. *Optimization*, 33(1):9–27, 1995.

[ZL17] Barret Zoph and Quoc Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.

[ZLJ$^+$20] Jingzhao Zhang, Hongzhou Lin, Stefanie Jegelka, Suvrit Sra, and Ali Jadbabaie. Complexity of finding stationary points of nonconvex nonsmooth functions. In *International Conference on Machine Learning*, pages 11173–11182. PMLR, 2020.

[ZLP$^+$22] Xiao Zhou, Yong Lin, Renjie Pi, Weizhong Zhang, Renzhe Xu, Peng Cui, and Tong Zhang. Model agnostic sample reweighting for out-of-distribution learning. In *International Conference on Machine Learning*, pages 27203–27221. PMLR, 2022.

[ZSP$^+$21] Miao Zhang, Steven W Su, Shirui Pan, Xiaojun Chang, Ehsan M Abbasnejad, and Reza Haffari. iDARTS: Differentiable architecture search with stochastic implicit gradients. In *International Conference on Machine Learning*, pages 12557–12566. PMLR, 2021.

A Empirical Studies

In this section we conduct selective empirical studies to validate the outperformance of algorithms proposed in this work. For BiO with LLSC, we validate the effectiveness and efficiency of our proposed algorithms—`RAF`^{2}`BA` and `PRAF`^{2}`BA`—by applying them to several machine learning tasks: hyperparameter optimization of logistic regression (20 News Group dataset) data, data hypercleaning (MNIST dataset), as well as a W-shaped synthetic example for minimax optimization. For BiO without LLSC, we compare `IGFM` with several baselines including AID with conjugate gradient [MDA15], ITD [JYL21], BGS [AM22b], BDA [LMY+20], BOME [LYW+22], and IA-GM [LLZZ21] in the application of adversarial training. Our experiments demonstrate that algorithms presented in this paper outperform established baseline algorithms such as BA, AID-BiO, ITD-BiO, PAID-BiO as well as `RAHGD`, `PRAHGD` proposed in the recent work [YLL+23], and (in synthetic minimax problem) the outperformance of our `PRAGDA` algorithm in comparison with IMCN proposed by [LLC22], exhibiting improved convergence rates.

A.1 Hyperparameter Optimization

The goal of *hyperparameter optimization* [GFPS20] is to find the optimal hyperparameter in minimizing the losses on the validation dataset. It can be cast to the BiO of form

$$
\begin{aligned}
&\min_{\lambda\in\mathbb{R}^p} \frac{1}{|\mathcal{D}_{\text{val}}|}\sum_{(x_i,y_i)\in\mathcal{D}_{\text{val}}} L(w^*(\lambda);x_i,y_i)\\
&\text{s.t.}\quad w^*(\lambda)=\arg\min_{w\in\mathbb{R}^{c\times p}} \frac{1}{|\mathcal{D}_{\text{tr}}|}\sum_{(x_i,y_i)\in\mathcal{D}_{\text{tr}}} L(w;x_i,y_i)+\frac{1}{2cp}\sum_{j=1}^{c}\sum_{k=1}^{p}\exp(\lambda_k)w_{jk}^2
\end{aligned}
$$

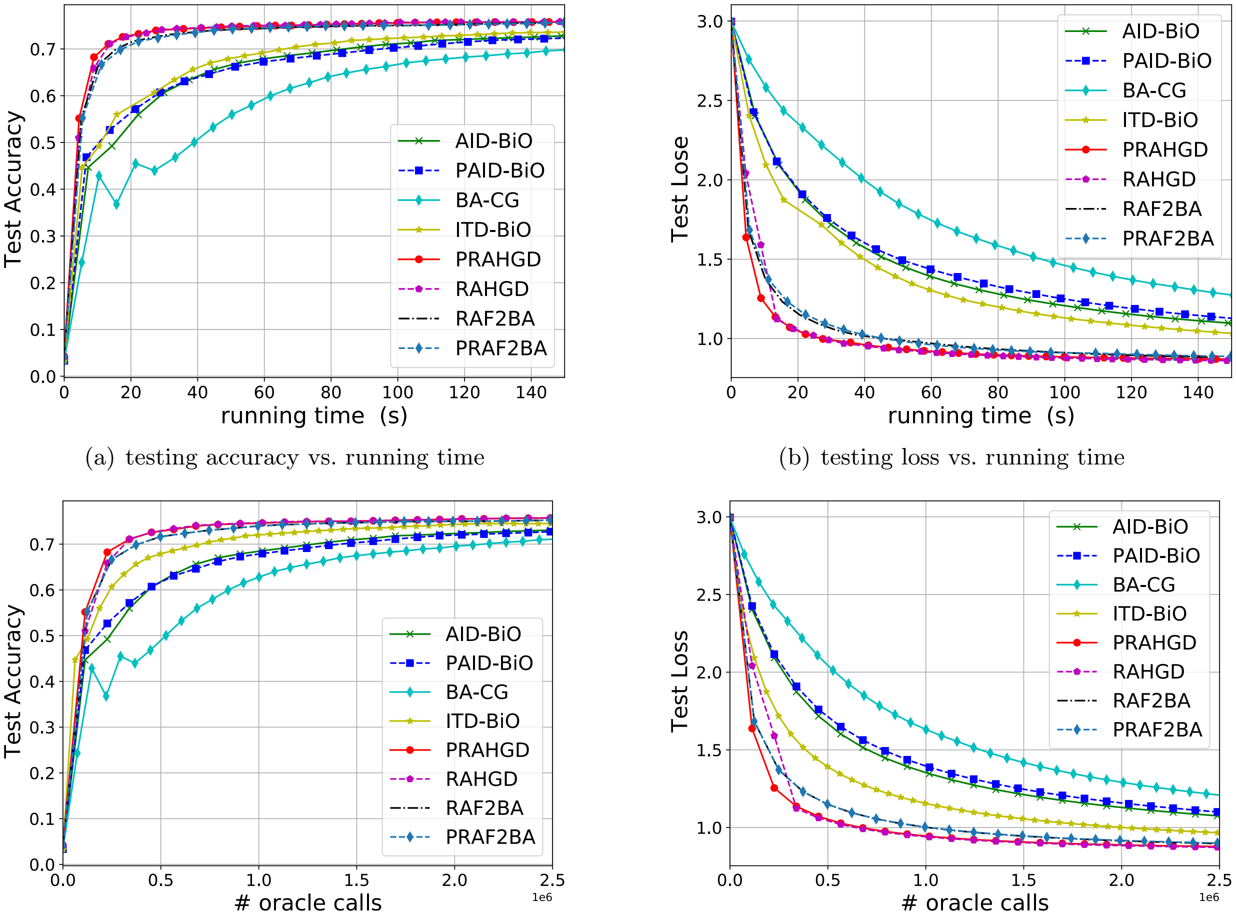
where $\mathcal{D}_{\text{tr}}=\{(x_i,y_i)\}$ is training dataset, $\mathcal{D}_{\text{val}}=\{(x_i,y_i)\}$ is validation dataset, L is cross-entropy loss function, $c=20$ is number of topics, and $p=130,170$ is dimension of features. As suggested in our theoretical part we use the *conjugate gradient* (CG) descent method to approximate the Hessian-inverse-vector product for `PRAHGD`, and fully first-order method for `(P)RAF`^{2}`BA`.

For a logistic regression problem on 20 News group dataset [GFPS20], we compare the performance of our algorithms with the baseline algorithms listed in Table 1. The dataset consists of 18,846 news items divided into 20 topics and features include 130,170 tf-idf sparse vectors. The data are divided into three parts: $|\mathcal{D}_{\text{tr}}|=5,657$ samples for training, $|\mathcal{D}_{\text{val}}|=5,657$ samples for validation and 7,532 samples for testing.

For algorithms listed in Figure 1, we tune both inner-loop and outer-loop learning rates from $\{0.001,0.01,0.1,1,10,100,1000\}$, where the iteration number of gradient descent or `AGD` steps are chosen from $\{5,10,30,50\}$, and the iteration number of CG step chosen from $\{5,10,30,50\}$. For BA-CG we choose the iteration number of gradient descent steps from $\{\lceil c(k+1)^{1/4}\rceil : c\in\{0.5,1,2,4\}\}$, as is adopted by [GW18]. For `RAF`^{2}`BA` and `PRAF`^{2}`BA`, we tune λ (in (5)) from $\{100,300,500,700\}$. The results are depicted in Figure 1, where we observe that our `RAHGD`, `PRAHGD`, `RAF`^{2}`BA` and `PRAF`^{2}`BA` evidently converge faster than rival algorithms.

A.2 Data Hypercleaning

In *data hypercleaning* [FDFP17, SCHB19] we have a dataset with label noise, and aim to train a model while cleaning up a subset of noisy data at limited time and/or cost. It is an application



(a) testing accuracy vs. running time

(b) testing loss vs. running time

(c) testing accuracy vs. number of oracle calls

(d) testing loss vs. number of oracle calls

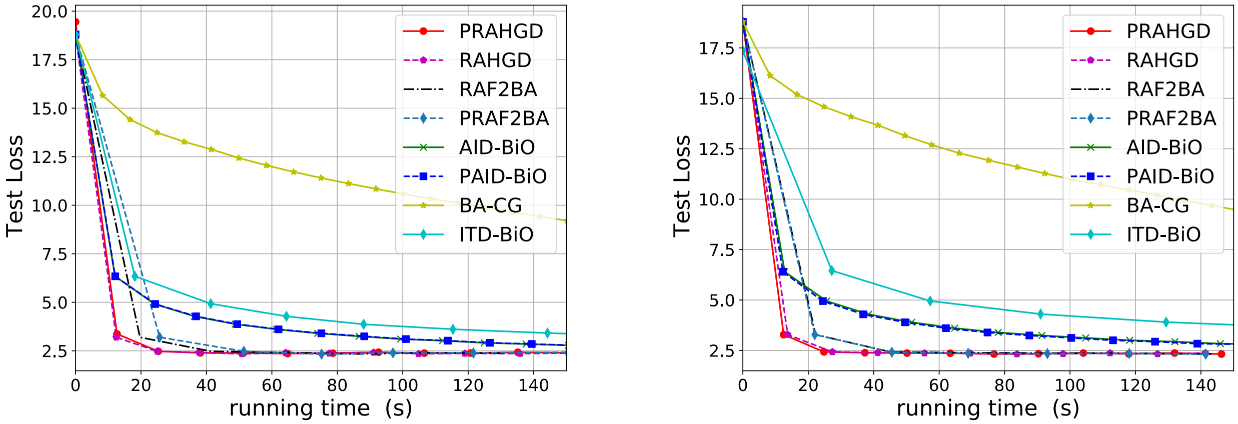
Figure 1. Comparison of a variety of bilevel algorithms on logistic regression on 20 Newsgroup dataset. Figures (a) and (b) depict the results of testing accuracy and testing loss vs. running time, respectively. Figures (c) and (d) depict the results of testing accuracy and testing loss vs. number of oracles calls, respectively.

example of BiO where one treats the cleaned data as the validation set and the remaining data as the training set:

$$\begin{aligned} &\min_{\lambda\in\mathbb{R}^{|\mathcal{D}_{\text{tr}}|}} \ f(W^*(\lambda),\lambda) \triangleq \frac{1}{|\mathcal{D}_{\text{val}}|}\sum_{(x_i,y_i)\in\mathcal{D}_{\text{val}}} -\log(y_i^\top W^*(\lambda)x_i) \\ &\text{s.t.} \quad W^*(\lambda) = \arg\min_{W\in\mathbb{R}^{d_y\times d_x}} \ g(W,\lambda) \triangleq \frac{1}{|\mathcal{D}_{\text{tr}}|}\sum_{(x_i,y_i)\in\mathcal{D}_{\text{tr}}} -\sigma(\lambda_i)\log(y_i^\top W x_i) + C_r\|W\|^2 \end{aligned} \tag{23}$$

where $\mathcal{D}_{\text{tr}} = \{(x_i,y_i)\}$ is training dataset, $\mathcal{D}_{\text{val}} = \{(x_i,y_i)\}$ is validation dataset, W is weight of the classifier, $\lambda_i \in \mathbb{R}$, $\sigma(\cdot)$ is the sigmoid function, and C_r is regularization parameter. We choose $C_r = 0.001$ following [SCHB19] and [JYL21].

We conducted an experiment on MNIST [LBBH98], which has $d_x = 785$ and $d_y = 10$ for problem (23). The training set contains $|\mathcal{D}_{\text{tr}}| = 20,000$ images, a significant portion of which have their labels randomly disrupted. We denote for image data the ratio of disrupted labels as the



(a) Corruption rate $p = 0.2$

(b) Corruption rate $p = 0.4$

Figure 2. Comparison of various bilevel algorithms for data hypercleaning at different corruption rates

corruption rate p. The validation set consists of $|\mathcal{D}_{\text{val}}| = 5,000$ images with correct labels; the testing set of 10,000 images.

The experimental results are depicted in Figure 2. Analogous to §A.1, we continue to use the CG to approximate the Hessian-inverse-vector product for PRAHGD, and fully first-order method for (P)RAF^{2}BA. For the BA algorithm proposed by [GW18], we also use CG descent method to compute the Hessian-inverse-vector product (note this was *not* specified in their work), namelly BA-CG in Figure 2. For all algorithms we tune the inner-loop and outer-loop learning rates from $\{0.001, 0.01, 0.1, 1, 10\}$, and the iteration number of CG step from $\{3, 6, 12, 24\}$. Except for BA, we choose for all algorithms the iteration number of gradient descent or AGD steps from $\{50, 100, 200, 500, 1000\}$; for BA algorithm, as adopted by [GW18] we choose the iteration number of gradient descent steps from $\{\lceil c(k+1)^{1/4} \rceil : c \in \{0.5, 1, 2, 4\}\}$. For RAF^{2}BA and PRAF^{2}BA we choose λ (in (5)) from $\{100, 300, 500, 700\}$. We observe that our RAHGD, PRAHGD, RAF^{2}BA and PRAF^{2}BA evidently converge faster than rival algorithms.

A.3 Adversarial Training

[BS11] proposed modeling adversarial training via BiO. In this model, the learner aims at finding the optimal parameter x, subject to data y being modified by an adversarial data provider.

Table 4. MSE (mean ± std) achieved by different algorithms on the abalone dataset in adversarial training.

Method	MSE
AID	1.781 ± 0.418
ITD	0.982 ± 0.015
BGS	0.995 ± 0.259
BDA	0.976 ± 0.014
BOME	0.999 ± 0.140
IA-GM	0.992 ± 0.013
IGFM (Ours)	**0.936 ± 0.015**

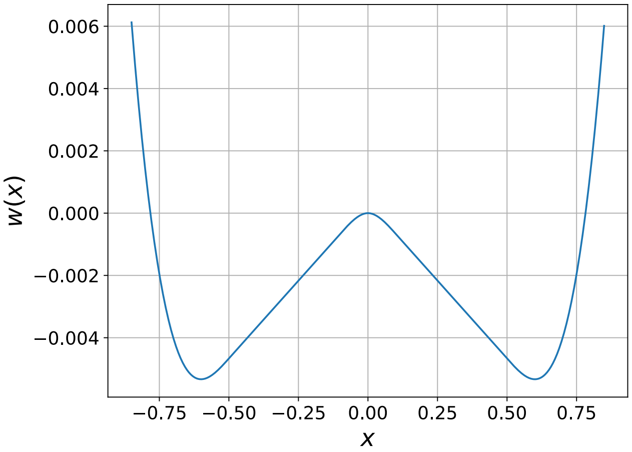


Figure 3: W-shape function [TSJ+18]

Like [BTTG20, WCJ+21, WHJ+22], we use least squares loss for both f and g as in Remark 4.2(iii). In the LL loss, we use a diagonal matrix M to assign different weights to each sample, and a ridge term $\|y - b\|_M^2$ to penalize the data provider when manipulating the original labels b. We set half the diagonal elements of M evenly in $[\sigma_{\min}^+, \sigma_{\max}]$ and the rest zero. We let $\lambda = 1$, $\sigma_{\max} = 1$ and $\sigma_{\min}^+ = 10^{-9}$. For BDA, we choose $s_u = s_l = 1$, $\alpha_k = \mu/(k+1)$ and tune μ from $\{0.1, 0.5, 0.9\}$ as [LMY+20]. For BOME, we choose the default option for ϕ_k and η from $\{0.9, 0.5, 0.1\}$ as [LYW+22]. For `IGFM`, we choose $\delta = 10^{-3}$ and tune θ from $\{10^{-1}, 10^{-2}, 10^{-3}\}$. For all algorithms, we tune the learning rates in $\{10^2, 10^1, 10^0, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. We run all the algorithms for 500 UL iterations, with 10 LL iterations per UL iteration. Table 4 compares the mean squared error (MSE), measured by the value of $\varphi(x)$, achieved by the algorithms on the `abalone` dataset from LIBSVM [CL11]. AID has poor performance because it requires taking the inverse of $\nabla_{yy}^2 g(x,y)$, which is ill-conditioned in this experiment. Among all the algorithms, the `IGFM` achieves the lowest mean value of MSE, and its variance is also maintained at a relatively low level.

A.4 W-Shaped Synthetic Minimax Example

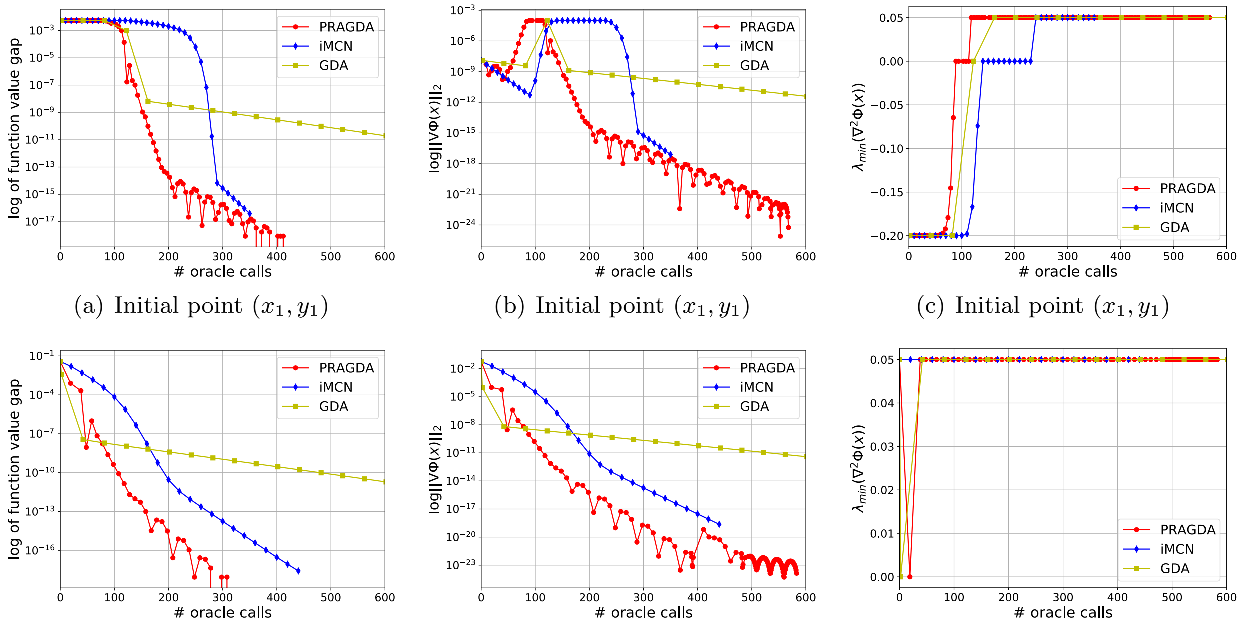
We construct the following nonconvex-strong-concave minimax problem

$$\min_{x \in \mathbb{R}^3} \max_{y \in \mathbb{R}^2} \ f(x,y) = w(x_3) - 10y_1^2 + x_1 y_1 - 5y_2^2 + x_2 y_2$$

where $x = [x_1, x_2, x_3]^\top$ and $y = [y_1, y_2]^\top$ and

$$w(x) = \begin{cases} \sqrt{\epsilon}(x + (L+1)\sqrt{\epsilon})^2 - \frac{1}{3}(x + (L+1)\sqrt{\epsilon})^3 - \frac{1}{3}(3L+1)\epsilon^{3/2} & x \leq -L\sqrt{\epsilon} \\ \epsilon x + \frac{\epsilon^{3/2}}{3} & -L\sqrt{\epsilon} < x \leq -\sqrt{\epsilon} \\ -\sqrt{\epsilon}x^2 - \frac{x^3}{3} & -\sqrt{\epsilon} < x \leq 0 \\ -\sqrt{\epsilon}x^2 + \frac{x^3}{3} & 0 < x \leq \sqrt{\epsilon} \\ -\epsilon x + \frac{\epsilon^{3/2}}{3} & \sqrt{\epsilon} < x \leq L\sqrt{\epsilon} \\ \sqrt{\epsilon}(x - (L+1)\sqrt{\epsilon})^2 + \frac{1}{3}(x - (L+1)\sqrt{\epsilon})^3 - \frac{1}{3}(3L+1)\epsilon^{3/2} & L\sqrt{\epsilon} < x \end{cases} \tag{24}$$

is the *W-shape-function* [TSJ+18] and we set $\epsilon = 0.01, L = 5$ in our experiment. $w(\cdot)$ is depicted in Figure 3.



(a) Initial point (x_1, y_1) (b) Initial point (x_1, y_1) (c) Initial point (x_1, y_1)

(d) Initial point (x_2, y_2) (e) Initial point (x_2, y_2) (f) Initial point (x_2, y_2)

Figure 4. A selection of empirical results with convergence measured by the function value gap, gradient norm and minimum eigenvalue of Hessian (in absolute value), applied on the task of synthetic minimax problem (13). The scale is in semi-log except for the absolute minimum Hessian eigenvalue.

It is straightforward to verify that $[x_0; y_0] = [[0, 0, 0]^\top; [0, 0]^\top]$ is a saddle point of $f(x, y)$. We propose our numerical experiments with the following two different initial points: $[x_1; y_1] = \left[[10^{-3}, 10^{-3}, 10^{-16}]^\top; [0, 0]^\top\right]$ and $[x_2, y_2] = \left[[0, 0, 1]^\top; [0, 0]^\top\right]$. Note $[x_1; y_1]$ is relatively close to initialization $[x_0; y_0]$ while $[x_2; y_2]$ relatively distant. We numerically compare our PRAGDA with IMCN [LLC22] and classical GDA [LJJ20a] algorithms. The results are depicted in Figure 4 where we adopted a grid search in choosing the inner-loop learning rates of AGD steps, GDA, and outer-loop learning rates of PRAGDA. The learning rates are tuned from $\{c \times 10^i : c \in \{1, 5\}, i \in \{1, 2, 3\}\}$ and momentum parameters from $\{c \times 0.1 : c \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}\}$.

We plot the results in Figure 4 the number of oracle calls versus $\varphi(x) - \varphi(x^*)$, $\|\nabla \varphi(x)\|$, and $\lambda_{\min}(\nabla^2 \varphi(x))$. Observing from the curves corresponding to initial point (x_2, y_2), all the three algorithms converge to the minimum when the initial point is relatively distant from the strict saddle point. However, our PRAGDA converges much faster than IMCN and GDA. When the initial point is relatively closer to the strict saddle Figure 4(b) depicts that the GDA algorithm can get stuck at saddles of the hyper-objective φ where the minimum eigenvalue of the Hessian is strictly negative. In contrast, our PRAGDA and IMCN can reach the points that admit positive Hessian minimum eigenvalues of the hyper-objective φ, whereas PRAGDA evidently converges faster than IMCN.

B Delegated Proofs of §2

B.1 Proofs of Basic Lemmas in §2.1

Proof of Lemma 1. Recall that

$$\nabla\varphi(x) = \nabla_x f(x, y^*(x)) - \nabla^2_{xy} g(x, y^*(x)) \big(\nabla^2_{yy} g(x, y^*(x))\big)^{-1} \nabla_y f(x, y^*(x))$$

We denote $\mathcal{H}_1(x) = \nabla_x f(x, y^*(x))$, $\mathcal{H}_2(x) = \nabla^2_{xy} g(x, y^*(x))$, $\mathcal{H}_3(x) = \big(\nabla^2_{yy} g(x, y^*(x))\big)^{-1}$ and $\mathcal{H}_4(x) = \nabla_y f(x, y^*(x))$, then

$$\nabla\varphi(x) = \mathcal{H}_1(x) - \mathcal{H}_2(x)\mathcal{H}_3(x)\mathcal{H}_4(x)$$

We first consider $\mathcal{H}_1(x), \mathcal{H}_2(x)$ and $\mathcal{H}_4(x)$. For any $x, x' \in \mathbb{R}^{d_x}$, we have

$$\|\mathcal{H}_1(x) - \mathcal{H}_1(x')\| \le \ell(\|x - x'\| + \|y^*(x) - y^*(x')\|) \le \ell(1+\kappa)\|x - x'\|$$

where we use triangle inequality in the first inequality and Lemma 2 in the second inequality.

We also have

$$\|\mathcal{H}_2(x) - \mathcal{H}_2(x')\| \le \rho(\|x - x'\| + \|y^*(x) - y^*(x')\|) \le \rho(1+\kappa)\|x - x'\|$$

and

$$\|\mathcal{H}_4(x) - \mathcal{H}_4(x')\| \le \ell(\|x - x'\| + \|y^*(x) - y^*(x')\|) \le \ell(1+\kappa)\|x - x'\|$$

We next consider $\mathcal{H}_3(x)$. For any $x, x' \in \mathbb{R}^{d_x}$, we have

$$\begin{aligned}
&\|\mathcal{H}_3(x) - \mathcal{H}_3(x')\| = \left\| \big(\nabla^2_{yy} g(x, y^*(x))\big)^{-1} - \big(\nabla^2_{yy} g(x', y^*(x'))\big)^{-1} \right\| \\
&\le \left\| \big(\nabla^2_{yy} g(x, y^*(x))\big)^{-1} \right\| \left\| \nabla^2_{yy} g(x', y^*(x')) - \nabla^2_{yy} g(x, y^*(x)) \right\| \left\| \big(\nabla^2_{yy} g(x', y^*(x'))\big)^{-1} \right\| \\
&\le \frac{1}{\mu^2}\rho \left(\|x - x'\| + \|y^*(x) - y^*(x')\|\right) \le \frac{\rho(1+\kappa)}{\mu^2}\|x - x'\|
\end{aligned}$$

We also have

$$\|\mathcal{H}_2(x)\| \le \ell \qquad \|\mathcal{H}_3(x)\| \le \frac{1}{\mu} \qquad \text{and} \quad \|\mathcal{H}_4(x)\| \le M$$

for any $x \in \mathbb{R}^{d_x}$. Then for any $x, x' \in \mathbb{R}^{d_x}$ we have

$$\begin{aligned}
&\|\nabla\varphi(x) - \nabla\varphi(x')\| \le \|\mathcal{H}_1(x) - \mathcal{H}_1(x')\| + \|\mathcal{H}_2(x)\mathcal{H}_3(x)\mathcal{H}_4(x) - \mathcal{H}_2(x')\mathcal{H}_3(x')\mathcal{H}_4(x')\| \\
&\le \ell(1+\kappa)\|x - x'\| + \|\mathcal{H}_2(x)\mathcal{H}_3(x)\mathcal{H}_4(x) - \mathcal{H}_2(x)\mathcal{H}_3(x)\mathcal{H}_4(x')\| \\
&\quad + \|\mathcal{H}_2(x)\mathcal{H}_3(x)\mathcal{H}_4(x') - \mathcal{H}_2(x)\mathcal{H}_3(x')\mathcal{H}_4(x')\| \\
&\quad + \|\mathcal{H}_2(x)\mathcal{H}_3(x')\mathcal{H}_4(x') - \mathcal{H}_2(x')\mathcal{H}_3(x')\mathcal{H}_4(x')\| \\
&\le \ell(1+\kappa)\|x - x'\| + \|\mathcal{H}_2(x)\|\|\mathcal{H}_3(x)\|\|\mathcal{H}_4(x) - \mathcal{H}_4(x')\| \\
&\quad + \|\mathcal{H}_2(x)\|\|\mathcal{H}_4(x')\|\|\mathcal{H}_3(x) - \mathcal{H}_3(x')\| \\
&\quad + \|\mathcal{H}_3(x')\|\|\mathcal{H}_4(x')\|\|\mathcal{H}_2(x) - \mathcal{H}_2(x')\| \\
&\le \ell(1+\kappa)\|x - x'\| + \frac{\ell^2}{\mu}(1+\kappa)\|x - x'\| + \frac{\ell\rho M}{\mu^2}(1+\kappa)\|x - x'\| + \frac{M\rho}{\mu}(1+\kappa)\|x - x'\| \\
&= \left(\ell + \frac{2\ell^2 + \rho M}{\mu} + \frac{\ell^3 + 2\rho\ell M}{\mu^2} + \frac{\rho\ell^2 M}{\mu^3}\right)\|x - x'\|
\end{aligned}$$

This completes the proof of the first part. For the second part see the detailed proof associated with [HJML22, Lemma 2.4]. □

Proof of Lemma 2. Recall that $y^*(x) = \arg\min_{y\in\mathbb{R}^{d_y}}\ g(x,y)$. The optimality condition leads to

$$\nabla_y g(x, y^*(x)) = 0$$

for each $x \in \mathbb{R}^{d_x}$. Taking a further derivative with respect to x on both sides and some algebra gives

$$\nabla^2_{yx} g(x, y^*(x)) + \nabla^2_{yy} g(x, y^*(x)) \frac{\partial y^*(x)}{\partial x} = 0$$

The smoothness and strong convexity of g in y immediately indicate

$$\frac{\partial y^*(x)}{\partial x} = -\left(\nabla^2_{yy} g(x, y^*(x))\right)^{-1} \nabla^2_{yx} g(x, y^*(x))$$

Thus we have

$$\left\|\frac{\partial y^*(x)}{\partial x}\right\| = \left\|\left(\nabla^2_{yy} g(x, y^*(x))\right)^{-1} \nabla^2_{yx} g(x, y^*(x))\right\| \le \frac{\ell}{\mu} = \kappa$$

where the inequality is based on the fact that $g(x, y)$ is ℓ-smooth with respect to x and y and μ-strongly convex with respect to y for any x. Therefore, we proved that $y^*(x)$ is κ-Lipschitz continuous. □

Proof of Lemma 3. Recall that

$$\nabla\varphi(x) = \nabla_x f(x, y^*(x)) - \nabla^2_{xy} g(x, y^*(x))\big(\nabla^2_{yy} g(x, y^*(x))\big)^{-1}\nabla_y f(x, y^*(x))$$

We define

$$\bar{\nabla}\varphi(x_k) = \nabla_x f(x_k, y_k) - \nabla^2_{xy} g(x_k, y_k)\big(\nabla^2_{yy} g(x_k, y_k)\big)^{-1}\nabla_y f(x_k, y_k)$$

then we have

$$\begin{aligned}
&\|\nabla\varphi(w_k) - \widehat{\nabla}\varphi(w_k)\|_2 = \|\nabla\varphi(w_k) - \bar{\nabla}\varphi(w_k) + \bar{\nabla}\varphi(w_k) - \widehat{\nabla}\varphi(w_k)\|_2 \\
&\le \|\nabla\varphi(w_k) - \bar{\nabla}\varphi(w_k)\|_2 + \|\bar{\nabla}\varphi(w_k) - \widehat{\nabla}\varphi(w_k)\|_2 \\
&\le \widetilde{L}\|y_k - y^*(w_k)\|_2 + \ell\left\|v_k - \big(\nabla^2_{yy} g(w_k, y_k)\big)^{-1}\nabla_y f(w_k, y_k)\right\|_2 \le \sigma
\end{aligned}$$

where we use the triangle inequality in the first inequality, Lemma 1 and Assumption 1(iii) in the second inequality and Condition 1 in the last inequality. □

B.2 Full Version of Lemma 5

Here we present the detailed expression of the upper bounds in Lemma 5. We refer readers to the cited reference for proof details.

Lemma 9. *[CMZ23, §B, §C] Suppose Assumption 1(i)–(iv) hold and set $\lambda \ge 2\kappa$, then*

(i) $|\mathcal{L}^*_\lambda(x) - \varphi(x)| \le D_0/\lambda$ for any $x \in \mathbb{R}^{d_x}$, where

$$D_0 = \left(M + \frac{M\ell}{2\mu}\right)\frac{M}{\mu} = \mathcal{O}(\kappa^2)$$

(ii) $\|\nabla \mathcal{L}_\lambda^*(x) - \nabla \varphi(x)\| \le D_1/\lambda$, *where*

$$D_1 = \left(\ell + \frac{2\rho\ell + M\rho}{2\mu} + \frac{M\rho\ell}{2\mu^2}\right)\frac{M}{\mu} = \mathcal{O}(\kappa^3) \tag{25}$$

(iii) $\mathcal{L}_\lambda^*(x)$ *is* L_λ*-Lipschitz, where*

$$L_\lambda = \ell + \frac{5\ell^2 + M\rho}{\mu} + \frac{2M\ell\rho + 2\ell^3}{\mu^2} + \frac{2M\ell^2\rho}{\mu^3} = \mathcal{O}(\kappa^3)$$

If we further suppose Assumption 1(v) holds, then

(i) $\left\|\nabla^2 \mathcal{L}_\lambda^*(x) - \nabla^2 \varphi(x)\right\| \le D_2/\lambda$ *for any* $x \in \mathbb{R}^{d_x}$, *where*

$$D_2 = 2\ell\left(1 + \frac{2\ell}{\mu}\right)^2\left(\frac{\ell}{\mu} + \frac{M\rho}{\mu^2}\right)^2 + \left(1 + \frac{\ell}{\mu}\right)^2\left(\frac{M\rho}{\mu} + \frac{M\ell\rho}{\mu^2} + \frac{M^2\nu}{2\mu^2} + \frac{M^2\rho^2}{2\mu^3}\right) = \mathcal{O}(\kappa^6)$$

(ii) $\mathcal{L}_\lambda^*(x)$ *is* ρ_λ*-Hessian Lipschitz, where*

$$\begin{aligned}\rho_\lambda = &\left(1 + \frac{4\ell}{\mu}\right)^2\left(3\rho + \frac{2\ell\rho}{\mu}\right) + \left(1 + \frac{\ell}{\mu}\right)^2\left(\frac{M\nu}{\mu} + \frac{M\rho^2}{\mu^2}\right) + \left(2 + \frac{5\ell}{\mu}\right)\left(1 + \frac{2\ell}{\mu}\right)\left(\frac{\ell\rho}{\mu} + \frac{M\rho^2}{\mu^2}\right)\\ &+ \frac{2\ell\rho}{\mu^2}\left(1 + \frac{\ell}{\mu}\right)^2\left(\ell + \frac{M\rho}{\mu}\right) + \frac{14\ell\rho}{\mu^2}\left(1 + \frac{2\ell}{\mu}\right)\left(\frac{\ell}{\mu} + \frac{M\rho}{\mu^2}\right) + \frac{50\ell^2}{\mu^3}\left(\frac{M\nu}{\mu} + \rho\right) = \mathcal{O}(\kappa^5)\end{aligned}$$

C Delegated Proofs of §4

C.1 Proof of Proposition 4.1

Proof of Proposition 4.1. By the definition that $y^* \in Y^*(x)$, we know that $g(x^*, y^*) = g^*(x^*)$. When $g(x, \cdot)$ is convex, we know that $Y^*(x)$ is also a convex set for any given x. Then the problem $\min_{y \in Y^*(x)} f(x, y)$ is a convex problem with respect to y, where a local minimum is also a global minimum. This indicates that $\varphi(x^*) = f(x^*, y^*)$. Finally, the first-order necessary optimality condition for a local minimum of $\varphi(x)$ implies that $\partial \varphi(x^*) = 0$ (Theorem 8.4 by [Cla17]).

□

C.2 Proof of Proposition 4.2

Proof of Proposition 4.2. We show that $Y^*(x)$ is Lipschitz, and then $\varphi(x)$ is also Lipschitz by Proposition 4.6.

Under Assumption 4.1, for any $y_1 \in Y^*(x_1)$, there exists $y_2 \in Y^*(x_2)$ such that

$$\begin{aligned}\alpha\|y_1 - y_2\| &\le \left\|\mathcal{G}_{1/L}(y_1; x_2) - \mathcal{G}_{1/L}(y_1; x_1)\right\|\\ &= L\left\|\mathcal{P}_\mathcal{Y}\left[y_1 - \frac{1}{L}\nabla_y g(x_2, y_1)\right] - \mathcal{P}_\mathcal{Y}\left[y_1 - \frac{1}{L}\nabla_y g(x_1, y_1)\right]\right\|\\ &\le \|\nabla_y g(x_2, y_1) - \nabla_y g(x_1, y_1)\| \le L\|x_1 - x_2\|\end{aligned}$$

where we use $\mathcal{G}_{1/L}(y_1; x_1) = 0$ [DL18] and Assumption 4.1 in the second line; the third line follows from the definition of the generalized gradient; the fourth line uses the non-expansiveness of projection operator by Corollary 2.2.3 in [N$^+$18]; and the last line uses the smoothness property of the LL function.

Under Assumption 4.2, for any $y_1 \in Y^*(x_1)$, there exists $y_2 \in Y^*(x_2)$ such that

$$\begin{aligned} &2\alpha\|y_1 - y_2\| \leq g(x_2, y_1) - g(x_2, y_2) \\ &\leq g(x_1, y_1) - g(x_1, y_2) + 2L\|x_1 - x_2\| \leq 2L\|x_1 - x_2\| \end{aligned}$$

where the last line uses $g(x_1, y_1) \leq g(x_1, y_2)$. □

C.3 Proof of Proposition 4.3

Proof of Proposition 4.3. We distinguish two different cases by whether we have $\widehat{y}_{[1]} = 0$.

When $\widehat{y}_{[1]} \neq 0$, we consider the problem given by

$$\min_{x \in \mathbb{R}, y \in Y^*(x)} \; y_{[1]}^2 - 2xy_{[1]} \qquad Y^*(x) = \arg\min_{y \in \mathbb{R}^2} \; (y_{[2]} - \widehat{y}_{[2]})^2$$

After adding regularization, we have $Y_\lambda^*(x) = \{\widehat{y}\}$ and $\varphi_\lambda(x) = \widehat{y}_{[1]}^2 - 2x\widehat{y}_{[1]}$.

When $\widehat{y}_{[1]} = 0$, we instead consider the problem given by

$$\min_{x \in \mathbb{R}, y \in Y^*(x)} \; (y_{[1]} + 1)^2 - 2x(y_{[1]} + 1) \qquad Y^*(x) = \arg\min_{y \in \mathbb{R}^2} \; (y_{[2]} - \widehat{y}_{[2]})^2$$

And after adding regularization we have $Y_\lambda^*(x) = \{0\}$ and $\varphi_\lambda(x) = 1 - 2x$. However, for both the two cases the original hyper-objective is the quadratic function $\varphi(x) = -x^2$.

□

C.4 Proof of Proposition 4.4

Proof of Proposition 4.4. Without loss of generality, we assume $y_0 = 0$. Let $d_y = q = 2K, \sigma = 1/\sqrt{q}$ and

$$f(x, y) = \frac{1}{2} \sum_{j=K+1}^{q} y_{[j]}^2 \qquad g(x, y) = \sigma^2 h_q\left(\frac{y}{\sigma}\right)$$

where $h_q(y)$ follows Definition 4.5. It is clear from the construction that both $f(x, \cdot)$, $g(x, \cdot)$ are convex and 1-gradient Lipschitz. Moreover, both of them are zero-chains. Then the property of zero-chain leads to

$$y_{k,[j]} = 0 \qquad \forall k + 1 \leq j \leq q \qquad 0 \leq k \leq K$$

Therefore $f(x, y_k)$ remains zero for all $0 \leq k \leq K$.

However, we know that $Y^*(x) = \{\sigma \mathbf{1}\}$. Therefore

$$\varphi(x) = \frac{1}{2} \sum_{j=K+1}^{q} \sigma^2 = \frac{K\sigma^2}{2} = \frac{1}{4}$$

which indicates that any first-order algorithm $\mathcal{A}$ has a constant sub-optimality gap. □

C.5 Proof of Proposition 4.5

Proof of Proposition 4.5. Without loss of generality, we assume $y_0 = 0$. Let $d_y = q = 2K$ and

$$f(x, y) = \sum_{j=K+1}^{q} \psi(y_{[j]}) \qquad g(x, y) = h_q(y)$$

where $h_q(y)$ follows Definition 4.6 and $\psi(y)$ the Huber function defined by

$$\psi(y) = \begin{cases} \beta y - \frac{1}{2}y^2 & y \geq \beta \\ \frac{1}{2}y^2 & -\beta < y < \beta \\ -\beta y + \frac{1}{2}y^2 & y \leq -\beta \end{cases}$$

Since $|\psi'(y)| \leq \beta$, we know $f(x, \cdot)$ is $(\sqrt{q}\beta)$-Lipschitz since

$$\left| \sum_{j=K+1}^{q} \psi(y_{[j]}) - \sum_{j=K+1}^{q} \psi(y'_{[j]}) \right| \leq \sum_{j=K+1}^{q} \left| \psi(y_{[j]}) - \psi(y'_{[j]}) \right| \leq \beta \sum_{j=K+1}^{q} \left| y_{[j]} - y'_{[j]} \right| \leq \beta\sqrt{q} \left\| y - y' \right\|$$

Let $\beta = 1/\sqrt{q}$ then $f(x, \cdot)$ is 1-Lipschitz. And $g(x, \cdot)$ is 1-Lipschitz on $\mathbb{B}(y^*(x))$.

Note that f always returns a zero subgradient at the origin, while g is a zero-chain. We have

$$y_{k,[j]} = 0 \qquad \forall k+1 \leq j \leq q \qquad 0 \leq k \leq K$$

Therefore $f(x, y_k)$ remains zero for all $0 \leq k \leq K$.

However, we know that $Y^*(x) = \{-\mathbf{1}/\sqrt{q}\}$. So it can be calculated that

$$\varphi(x) = \sum_{j=K+1}^{q} \psi\left(-\frac{1}{\sqrt{q}}\right) = -\frac{K}{2q} = -\frac{1}{4}$$

indicating that any first-order algorithm $\mathcal{A}$ has a constant sub-optimality gap. □

We remark that projection onto the ball centered at the origin $\mathbb{B}(0)$ will not produce additional nonzero entries. Therefore, the possible projection operation in the algorithm will not distort the zero-chain structure.

C.6 Proof of Proposition 4.6

Proof of Proposition 4.6. Note that we can replace sup and inf with max and min in Definition D.3 due to the compactness of $Y^*(x)$. Below we prove each part of the proposition, item-by-item:

Proof of (i). Since $Y^*(x_1), Y^*(x_2)$ are nonempty compact sets, we can pick

$$y_1 \in \arg\min_{y \in Y^*(x_1)} f(x_1, y) \qquad y_2 \in \arg\min_{y \in Y^*(x_2)} f(x_2, y)$$

Then the Lipschitz continuity of $Y^*(x)$ implies there exist $y'_1 \in Y^*(x_1)$ and $y'_2 \in Y^*(x_2)$ such that

$$\varphi(x_1) - \varphi(x_2) \leq f(x_1, y'_1) - f(x_2, y_2) \leq C_f \left(\|x_1 - x_2\| + \|y_2 - y'_1\| \right) \leq (\kappa + 1)C_f \|x_1 - x_2\|$$

$$\varphi(x_2) - \varphi(x_1) \leq f(x_2, y'_2) - f(x_1, y_1) \leq C_f \left(\|x_1 - x_2\| + \|y_1 - y'_2\| \right) \leq (\kappa + 1)C_f \|x_1 - x_2\|$$

This establishes the Lipschitz continuity of φ.

Proof of (ii). It suffices to bound the following term for any x_1, x_2

$$\max \left\{ \underbrace{\max_{y_2 \in Y^*(x_2)} \min_{y_1 \in Y^*(x_1)} \|y_1 - y_2\|}_{\text{(I)}}, \underbrace{\max_{y_1 \in Y^*(x_1)} \min_{y_2 \in Y^*(x_2)} \|y_1 - y_2\|}_{\text{(II)}} \right\} \tag{26}$$

Without loss of generality, we assume $C_f = 1$, otherwise we can scale $f(x,y)$ by C_f to prove the result. We let $f(x,y) = -\min_{y_1 \in Y^*(x_1)} \|y - y_1\|$, then

$$\text{(I)} = \varphi(x_1) - \varphi(x_2) \leq C_\varphi \|x_1 - x_2\|$$

Next, we let $f(x,y) = \max_{y_1 \in Y^*(x_1)} \|y - y_1\|$, then

$$\text{(II)} \leq \varphi(x_2) - \varphi(x_1) \leq C_\varphi \|x_1 - x_2\|$$

Together, recalling the definition of (I) and (II) in (26), we know that

$$\operatorname{dist}(Y^*(x_1), Y^*(x_2)) \leq C_\varphi \|x_1 - x_2\| \qquad \forall x_1, x_2 \in \mathbb{R}^d$$

Proposition 4.6(iii) and Proposition 4.6(iv) replace the global Lipschitz continuity with local Lipschitz continuity. The proofs are similar, with additional care for the local argument.

Proof of (iii). We use $\mathcal{N}_\delta(\cdot)$ to denote the open neighborhood ball with radius δ. For a vector z, we define $\mathcal{N}_\delta(z) \triangleq \{z' : \|z' - z\| < \delta\}$. For a set S, we define $\mathcal{N}_\delta(S) \triangleq \{z' : \operatorname{dist}(z', S) < \delta\}$. For a given $x_1 \in \mathbb{R}^d$ and any $y \in Y^*(x_1)$, the local Lipschitz continuity of $f(\cdot,\cdot)$ implies that there exists $\delta_y > 0$ and $L_y > 0$ such that $f(\cdot,\cdot)$ is L_y-Lipschitz in $\mathcal{N}_{\delta_y}(x_1) \times \mathcal{N}_{\delta_y}(y)$. Note that the set $S \triangleq \bigcup_y \left\{ \mathcal{N}_{\delta_y}(x_1) \times \mathcal{N}_{\delta_y}(y) \right\}$ forms an open cover of the set $x_1 \times Y^*(x_1)$. The compactness of set $Y^*(x_1)$ guarantees the existence of a finite subcover $\bigcup_{k=0}^n \left\{ \mathcal{N}_{\delta_{y_k}}(x_1) \times \mathcal{N}_{\delta_{y_k}}(y_k) \right\}$. Therefore, we can conclude that there exists $\delta_1 > 0$ such that $f(\cdot,\cdot)$ is L_1-Lipschitz in the neighborhood $\mathcal{N}_{\delta_1}(x_1) \times \mathcal{N}_{\delta_1}(Y^*(x_1))$, where $L_1 = \max_k L_{y_k}$.

Next, the local Lipschitz continuity of $Y^*(\cdot)$ implies the existence of $\delta_2 > 0$ and $L_2 > 0$ such that $Y^*(\cdot)$ is L_2-Lipschitz in $\mathcal{N}_{\delta_2}(x_1)$. Take $\delta = \min\{\delta_1, \delta_2, \delta_1 / L_2\}$. The choice of δ ensures $(x_2, y_2) \in \mathcal{N}_{\delta_1}(x_1) \times \mathcal{N}_{\delta_1}(Y^*(x_1))$ for any $x_2 \in \mathcal{N}_\delta(x_1)$ and $y_2 \in Y^*(x_2)$. For any $x_2 \in \mathcal{N}_\delta(x_1)$, we pick

$$y_1 \in \arg\min_{y \in Y^*(x_1)} f(x_1, y) \qquad y_2 \in \arg\min_{y \in Y^*(x_2)} f(x_2, y)$$

The Lipschitz continuity of $f(\cdot,\cdot)$ in $\mathcal{N}_{\delta_1}(x_1) \times \mathcal{N}_{\delta_1}(Y^*(x_1))$ and the Lipschitz continuity of $Y^*(\cdot)$ in $\mathcal{N}_{\delta_2}(x_1)$ implies there exist $y_1' \in Y^*(x_1)$ and $y_2' \in Y^*(x_2)$ such that

$$\varphi(x_1) - \varphi(x_2) \leq f(x_1, y_1') - f(x_2, y_2) \leq L_1 \left(\|x_1 - x_2\| + \|y_2 - y_1'\| \right) \leq (L_2 + 1) L_1 \|x_1 - x_2\|$$

$$\varphi(x_2) - \varphi(x_1) \leq f(x_2, y_2') - f(x_1, y_1) \leq L_1 \left(\|x_1 - x_2\| + \|y_1 - y_2'\| \right) \leq (L_2 + 1) L_1 \|x_1 - x_2\|$$

hold for any $x_2 \in \mathcal{N}_\delta(x_1)$, implying the locally Lipschitz property of $\varphi(\cdot)$.

Proof of (iv). We again use the function $f(x, y)$ in the proof of **(ii)** to bound (I) and (II) defined in (26) Let $f(x, y) = -\min_{y_1 \in Y^*(x_1)} \|y - y_1\|$, then there exist $\delta_1 > 0$ and $L_1 > 0$ such that

$$(\mathrm{I}) = \varphi(x_1) - \varphi(x_2) \leq L_1 \|x_1 - x_2\| \qquad \forall \|x_1 - x_2\| \leq \delta_1$$

Let $f(x, y) = \max_{y_1 \in Y^*(x_1)} \|y - y_1\|$, then there exist $\delta_2 > 0$ and $L_2 > 0$ such that

$$(\mathrm{II}) \leq \varphi(x_2) - \varphi(x_1) \leq L_2 \|x_1 - x_2\| \qquad \forall \|x_1 - x_2\| \leq \delta_2$$

Together, taking $\delta = \min\{\delta_1, \delta_2\}$ and $L = \max\{L_1, L_2\}$ and recalling the definition of (I) and (II) in (26), we can show that there exists some $\delta > 0$ such that it holds

$$\mathrm{dist}(Y^*(x_1), Y^*(x_2)) \leq L \|x_1 - x_2\| \qquad \forall \|x_1 - x_2\| \leq \delta$$

which implies the local Lipschitz property of $Y^*(\cdot)$.

□

D Miscellaneous for BiO without LLSC

D.1 Backgrounds

Here we provide some necessary backgrounds to readers

Constrained Optimization. To tackle the possible constraint in y, we introduce the definitions of projection and generalized gradient [N+18] as follows.

Definition D.1 (Projection)**.** *We define the* projection onto a set $\mathcal{Y}$ *by* $\mathcal{P}_{\mathcal{Y}}(\cdot) \triangleq \arg\min_{y \in \mathcal{Y}} \|y - \cdot\|$.

Definition D.2 (Generalized Gradient)**.** *For a L-gradient Lipschitz function $g(x, y)$ with $y \in \mathcal{Y}$, we define the* generalized gradient with respect to y *by* $\mathcal{G}_\eta(y; x) \triangleq (y - \mathcal{P}_{\mathcal{Y}}(y - \eta \nabla_y g(x, y)))/\eta$ *with some* $0 < \eta \leq 1/L$.

Note that the generalized gradient reduced to $\nabla_y g(x, y)$ when $\mathcal{Y} = \mathbb{R}^{d_y}$.

Set-Valued Analysis. A classic notion of distance in set-valued analysis is the Hausdorff distance [RW09], formally defined as follows.

Definition D.3 (Hausdorff Distance)**.** *The* Hausdorff distance *between two sets S_1, S_2 is defined as*

$$\mathrm{dist}\,(S_1, S_2) = \max \left\{ \sup_{x_1 \in S_1} \inf_{x_2 \in S_2} \|x_1 - x_2\|, \sup_{x_2 \in S_2} \inf_{x_1 \in S_1} \|x_1 - x_2\| \right\}$$

This allows us to define the Lipschitz continuity of set-valued mappings as follows.

Definition D.4. *We call a set-valued mapping $S(x) : \mathbb{R}^{d_1} \rightrightarrows \mathbb{R}^{d_2}$* locally Lipschitz *if for any $x \in \mathbb{R}^{d_1}$, there exists $\delta > 0$ and $L > 0$ such that for any $x' \in \mathbb{R}^{d_1}$ satisfying $\|x' - x\| \leq \delta$, we have* $\mathrm{dist}(S(x), S(x')) \leq L\|x - x'\|$. *We call $S(x)$* Lipschitz *if we can let $\delta \to \infty$.*

Note that the above definition generalizes the Lipschitz continuity for a single-valued mapping.

Nonsmooth Analysis. The following Clarke subdifferential [Cla90] generalizes both the gradients of differentiable functions and the subgradients of convex functions.

Definition D.5 (Clarke Subdifferential)**.** *The* Clarke subdifferential *of a locally Lipschitz function $h(x) : \mathbb{R}^d \to \mathbb{R}$ at a point $x \in \mathbb{R}^d$ is defined by*

$$\partial h(x) \triangleq \operatorname{Conv}\left\{ s \in \mathbb{R}^d : \exists x_k \to x, \nabla h(x_k) \to s \quad \text{s.t. } \nabla h(x_k) \textit{ exists for all } k \right\}$$

It can be proved that finding a point with a small Clarke subdifferential is generally intractable for a nonsmooth nonconvex function [ZLJ$^+$20]. So we need to consider the following relaxed definition of stationarity for non-asymptotic analysis in nonsmooth nonconvex optimization [ZLJ$^+$20, TZS22, DDL$^+$22, JKL$^+$23, KS21, LZJ22, CMO23, KS24].

Definition D.6 (Approximate Goldstein Stationary Point)**.** *Given a locally Lipschitz function $h(x) : \mathbb{R}^d \to \mathbb{R}$, we call $x \in \mathbb{R}^d$ a (δ, ε)-*Goldstein stationary point *if $\min\{\|s\| : s \in \partial_\delta h(x)\} \le \varepsilon$, where $\partial_\delta h(x) \triangleq \operatorname{Conv}\left\{\cup_{x' \in \mathbb{B}_\delta(x)} \partial h(x')\right\}$ is the* Goldstein subdifferential *[Gol77].*

D.2 Limitations of Value-Function Approach

In contrast to the hyper-objective approach adopted in this section that pursues a UL stationary point such that $\|\nabla \varphi(x)\| \le \varepsilon$, existing non-asymptotic analysis [LYW$^+$22, SJGL22] for BiO without LLSC relies on following value-function reformulation for Problem (1)

$$\min_{x \in \mathbb{R}^{d_x}, y \in \mathbb{R}^{d_y}} f(x, y) \qquad \text{s.t.} \quad g(x, y) - g^*(x) \le 0 \tag{27}$$

These value-function approaches show convergence to the following KKT points.

Definition D.7 (KKT point)**.** *Suppose that $g^*(x)$ is Clarke subdifferentiable. We call (x, y) an ε-*KKT point *of Problem (27) if there exists a scalar $\lambda \ge 0$ such that*

(i) (Stationary in x) $\|\nabla_x f(x, y) + \lambda(\nabla_x g(x, y) - \partial g^*(x))\| \le \varepsilon$

(ii) (Stationary in y) $\|\nabla_y f(x, y) + \lambda \nabla_y g(x, y)\| \le \varepsilon$

(iii) (Feasibility) $g(x, y) - g^*(x) \le \varepsilon$

(iv) (Complementary Slackness) $|\lambda(g(x, y) - g^*(x))| \le \varepsilon$

We call (x, y) a KKT point if $\varepsilon = 0$.

Remark D.1. *In Definition D.7 we assume that $g^*(x)$ is Clarke differentiable. It can be easily satisfied under some mild conditions. For instance, when $g(x, y)$ is L-gradient Lipschitz, $g^*(x)$ is provably L-weakly concave, and thus Clarke differentiable [RW09]. In the unconstrained case that $\mathcal{Y} = \mathbb{R}^{d_y}$, under LLSC or more generally under Assumption 4.1, $g^*(x)$ is provably differentiable [NSH$^+$19] and the Clarke subdifferential $\partial g^*(x)$ reduces to the classical gradient $\nabla g^*(x)$.*

Unfortunately, classical constraint qualifications provably fail for the value-function-based reformulation [YZ95]. For this reason, we can easily construct a BiO instance whose KKT points do not contain the optimal solution even under LLSC.

Example D.1. *Consider a BiO instance given by*

$$\min_{x\in\mathbb{R},y\in\mathbb{R}} \; -xy \qquad s.t. \; (x+y-2)^2 \le 0$$

where the LL function is strongly convex in y. For this example

(i) The stationary point of $\varphi(x)$ is exactly the global solution x^*

(ii) However, the KKT points by Definition D.7 do not include any solution to this problem

Proof. We know that the LL constraint is $y = 2-x$, so the problem is equivalent to $\min_{x\in\mathbb{R}} x^2-2x$ with the unique solution $(x^*,y^*) = (1,1)$. However, if we rewrite the problem by

$$\min_{x\in\mathbb{R},y\in\mathbb{R}} \; -xy \qquad \text{s.t. } (x+y-2)^2 \le 0$$

The KKT condition is

$$\begin{cases} y-2\lambda(x+y-2)=0 \\ x-2\lambda(x+y-2)=0 \\ \lambda(x+y-2)^2=0 \\ (x+y-2)^2 \le 0 \\ \lambda \ge 0 \end{cases}$$

When $\lambda > 0$ there is no (x,y) that satisfies the KKT condition. When $\lambda = 0$, the KKT condition is only satisfied by $(x,y) = (0,0)$, but it is not the solution to this problem. □

One may argue that when relaxing the goal into finding an ε-KKT point, Slater's constraint qualification can be satisfied since we allow the constraint $g(x,y)-g^*(x) \le 0$ to be violated slightly. However, we give a concrete example indicating that an ε-KKT point may be far away from the solution set, even when the hyper-objective $\varphi(x)$ is strongly convex.

Example D.2. *Given $0 < \varepsilon \le 1$. Suppose $\varphi(x)$ is μ-strongly convex with a unique solution x^*.*

(i) Whenever a given point x satisfies $\|\nabla\varphi(x)\| \le \varepsilon$, we have $\|x-x^*\| \le \varepsilon/\mu$

(ii) However, there exists a BiO instance with a convex LL function such that the resulting $\varphi(x)$ is strongly convex, but there is an infinite number of 2ε-stationary points (x,y) by Definition D.7 such that $\|x-x^*\| = 1$

Proof. Below we prove the two parts in order.

Proof of (i). Strong convexity ensures that $\mu\|x-x^*\| \le \|\nabla\varphi(x)\|$

Proof of (ii). Consider the bilevel problem given by

$$\min_{x\in\mathbb{R},y\in\mathbb{R}} \; x^2-2\varepsilon xy \qquad \text{s.t. } y \in \arg\min_{y\in\mathbb{R}} \; \varepsilon^3 y^2$$

where the LL problem is convex in y and the global solution is $x^*=0$. It can be verified that $(x,y) = (1,\varepsilon^{-1})$ is an ε-KKT point with any multiplier satisfying $0 < \lambda \le 1$ by

$$\begin{cases} g(x,y)-g^*(x) = \varepsilon^3 y^2 = \varepsilon \\ |\nabla_x f(x,y) + \lambda\left(\nabla_x g(x,y) - \nabla g^*(x)\right)| = 2(x-\varepsilon y) = 0 \\ |\nabla_y f(x,y) + \lambda \nabla_y g(x,y)| = 2\left(\varepsilon x - \lambda\varepsilon^3 y\right) \le 2\varepsilon \end{cases}$$

But we know that $\|x-x^*\| = 1$. □