

DIFFERENTIALLY PRIVATE BILEVEL OPTIMIZATION

Anonymous authors

Paper under double-blind review

ABSTRACT

We present differentially private (DP) algorithms for bilevel optimization, a problem class that received significant attention lately in various machine learning applications. These are the first DP algorithms for this task that are able to provide any desired privacy, while also avoiding Hessian computations which are prohibitive in large-scale settings. Under the well-studied setting in which the upper-level is not necessarily convex and the lower-level problem is strongly-convex, our proposed gradient-based (ϵ, δ) -DP algorithm returns a point with hypergradient norm at most $\tilde{O}((\sqrt{d_{\text{up}}}/\epsilon n)^{1/2} + (\sqrt{d_{\text{low}}}/\epsilon n)^{1/3})$ where n is the dataset size, and $d_{\text{up}}/d_{\text{low}}$ are the upper/lower level dimensions. Our analysis covers constrained and unconstrained problems alike, accounts for mini-batch gradients, and applies to both empirical and population losses.

1 INTRODUCTION

Bilevel optimization is a fundamental framework for solving optimization objectives of hierarchical structure, in which constraints are defined themselves by an auxiliary optimization problem. Formally, it is defined as

$$\begin{aligned} & \text{minimize}_{\mathbf{x} \in \mathcal{X}} F(\mathbf{x}) := f(\mathbf{x}, \mathbf{y}^*(\mathbf{x})) & (\text{BO}) \\ & \text{subject to } \mathbf{y}^*(\mathbf{x}) \in \arg \min_{\mathbf{y}} g(\mathbf{x}, \mathbf{y}), \end{aligned}$$

where $F : \mathbb{R}^{d_x} \rightarrow \mathbb{R}$ is referred to as the hyperobjective, $f : \mathbb{R}^{d_x} \times \mathbb{R}^{d_y} \rightarrow \mathbb{R}$ as the upper-level (or outer) objective, and $g : \mathbb{R}^{d_x} \times \mathbb{R}^{d_y} \rightarrow \mathbb{R}$ as the lower-level (or inner) objective. While bilevel optimization is well studied for over half a century (Bracken & McGill, 1973), it has recently received significant attention due to its diverse applications in machine learning (ML). These include hyperparameter tuning (Bengio, 2000; Maclaurin et al., 2015; Franceschi et al., 2017; 2018; Lorraine et al., 2020), meta-learning (Andrychowicz et al., 2016; Bertinetto et al., 2018; Rajeswaran et al., 2019; Ji et al., 2020), neural architecture search (Liu et al., 2018), invariant learning (Arjovsky et al., 2019; Jiang & Veitch, 2022), and data reweighting (Grangier et al., 2023; Fan et al., 2024; Pan et al., 2024). In these applications, both the upper and lower level objectives in (BO) typically represent some loss over data, and are given by empirical risk minimization (ERM) problems with respect to some dataset $S = \{\xi_1, \dots, \xi_n\} \in \Xi^n$.¹

$$f(\mathbf{x}, \mathbf{y}) := f_S(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^n f(\mathbf{x}, \mathbf{y}, \xi_i), \quad g(\mathbf{x}, \mathbf{y}) := g_S(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^n g(\mathbf{x}, \mathbf{y}, \xi_i), \quad (\text{ERM})$$

often serving as empirical proxies of the stochastic (population) objectives with respect to a distribution \mathcal{P} supported on Ξ :

$$f(\mathbf{x}, \mathbf{y}) := f_{\mathcal{P}}(\mathbf{x}, \mathbf{y}) = \mathbb{E}_{\xi \sim \mathcal{P}}[f(\mathbf{x}, \mathbf{y}; \xi)], \quad g(\mathbf{x}, \mathbf{y}) := g_{\mathcal{P}}(\mathbf{x}, \mathbf{y}) = \mathbb{E}_{\xi \sim \mathcal{P}}[g(\mathbf{x}, \mathbf{y}; \xi)]. \quad (\text{Pop})$$

In this work, we study bilvel optimization under differential privacy (DP) (Dwork et al., 2006). As ML models are deployed in an ever-growing number of applications, protecting the privacy of the

¹It is possible for the datasets with respect to f and g to be distinct (e.g., validation and training data), perhaps of different sizes. We will assume without loss of generality that S is the entire dataset, and thus n is the total number of samples. Concretely, letting $f(\cdot; \xi_i) = 0$ or $g(\cdot; \xi_i) = 0$ for certain indices in order to exclude corresponding data points from either objective, will not affect our results.

054 data on which they are trained is a major concern, and DP has become the gold-standard for privacy
 055 preserving ML (Abadi et al., 2016). Accordingly, DP optimization is extensively studied, with
 056 a vast literature focusing both on empirical and stochastic objectives under various assumptions
 057 (Chaudhuri et al., 2011; Kifer et al., 2012; Bassily et al., 2014; Wang et al., 2017; Bassily et al.,
 058 2019; Wang et al., 2019; Feldman et al., 2020; Tran & Cutkosky, 2022; Gopi et al., 2022; Arora
 059 et al., 2023; Carmon et al., 2023; Ganesh et al., 2024; Lowy et al., 2024).

060 Nonetheless, to the best of our knowledge, no first-order algorithm (i.e., which uses only gradient
 061 queries) that solves bilevel optimization problems under DP, is known to date. This is no coin-
 062 cidence: until recently, no first-order methods with finite time guarantees were known even for
 063 non-private bilevel problems. This follows the fact (Ghadimi & Wang, 2018, Lemma 2.1) that under
 064 mild regularity assumptions, the so-called hypergradient takes the form:

$$065 \quad \nabla F(\mathbf{x}) = \nabla_x f(\mathbf{x}, \mathbf{y}^*(\mathbf{x})) - \nabla_{xy}^2 g(\mathbf{x}, \mathbf{y}^*(\mathbf{x})) [\nabla_{yy}^2 g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))]^{-1} \nabla_y f(\mathbf{x}, \mathbf{y}^*(\mathbf{x})). \quad (1)$$

066
 067 Consequently, directly applying a “gradient” method to F requires inverting Hessians of the lower
 068 level problem at each time step, thus limiting applicability in contemporary high-dimensional appli-
 069 cations. Following various approaches to tackle this challenge (see Section 1.2), recent break-
 070 throughs were finally able to provide fully first-order methods for bilevel optimization with non-
 071 asymptotic guarantees (Liu et al., 2022; Kwon et al., 2023; Yang et al., 2023; Chen et al., 2024).
 072 These recent algorithmic advancements show promising empirical results in large scale applica-
 073 tions, even up to the LLM scale of $\sim 10^9$ parameters (Pan et al., 2024). We therefore make use of
 074 these techniques for the sake of private optimization. As we will see, our privacy analysis requires
 075 overcoming some subtle challenges due to privacy leaking between the inner and outer problems,
 076 since $\nabla F(\mathbf{x})$ itself depends on $\mathbf{y}^*(\mathbf{x})$ (seen in Eq. (1)).

077 The only prior method we are aware of for DP bilevel optimization was recently proposed by Chen
 078 & Wang (2024), which falls short in two main aspects. First, their algorithm only provides *some*
 079 privacy guarantee which cannot be controlled by the user. Moreover, it requires inverting local
 080 Hessians at each step, which significantly limits scalability; see Section 1.2 for further discussion.

081 1.1 OUR CONTRIBUTIONS

082
 083 We present DP algorithms that solve bilevel optimization problems whenever the the upper level is
 084 smooth but not necessarily convex, and the lower level problem is smooth and strongly-convex. To
 085 the best of our knowledge, these are the first algorithms to do so using only first-order (i.e. gradient)
 086 queries of the upper- and lower-level objectives, and that can ensure any desired privacy $\epsilon, \delta > 0$.
 087 Our contributions can be summarized as follows:

- 088 • **Bilevel ERM Algorithm (Theorem 3.1):** We present a (ϵ, δ) -DP first-order algorithm for the
 089 bilevel ERM problem (BO/ERM) that outputs with high probability a point with hypergradient
 090 norm bounded by

$$091 \quad \|\nabla F_S(\mathbf{x})\| = \tilde{O} \left(\left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} \right).$$

092
 093 Our algorithm also adapts to the case where $\mathcal{X} \subsetneq \mathbb{R}^{d_x}$ is a non-trivial constraint set, which is
 094 common in certain applications.² In the constrained setting, we obtain the same guarantee as
 095 above in terms of the projected hypergradient (see Section 2 for details).

- 096 • **Mini-batch Bilevel ERM Algorithm (Theorem 4.1):** Aiming for a more practical algorithm,
 097 we design a variant of our previous algorithm that relies on mini-batch gradients. For the bilevel
 098 ERM problem (BO/ERM), given any batch sizes $b_{\text{in}}, b_{\text{out}} \in \{1, \dots, n\}$ for sampling gradients
 099 of the inner/outer problems respectively, our algorithm ensures (ϵ, δ) -DP and outputs with high
 100 probability a point with hypergradient norm bounded by

$$101 \quad \|\nabla F_S(\mathbf{x})\| = \tilde{O} \left(\left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} + \frac{1}{b_{\text{out}}} \right). \quad (2)$$

102
 103
 104
 105
 106
 107 ²For instance, in data reweighting \mathcal{X} is the probability simplex; in hyperparameter tuning it is the hyperpa-
 rameter space, which is typically constrained.

Notably, Eq. (2) is independent of the inner-batch size, yet depends on the outer-batch size, which coincides with known results for “single”-level constrained nonconvex optimization (Ghadimi et al., 2016) (see Remark 4.2 for further discussion). Our mini-batch algorithm is also applicable in the constrained setting $\mathcal{X} \subsetneq \mathbb{R}^{d_x}$ with the same guarantee in terms of projected hypergradient.

- **Population loss guarantees (Theorem 5.1):** We further provide guarantees for stochastic objectives. In particular, we show that for the population bilevel problem (BO/Pop), our (ϵ, δ) -DP algorithm outputs with high probability a point with hypergradient norm bounded by

$$\|\nabla F_{\mathcal{P}}(\mathbf{x})\| = \tilde{\mathcal{O}} \left(\left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + \left(\frac{d_x}{n} \right)^{1/2} + \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} \right),$$

with an additional additive $1/b_{\text{out}}$ factor in the mini-batch setting.

1.2 RELATED WORK

Bilevel optimization was introduced by Bracken & McGill (1973), and grew into a vast body of work, with classical results focusing on asymptotic guarantees for specific problem structures (Anandalingam & White, 1990; Ishizuka & Aiyoshi, 1992; White & Anandalingam, 1993; Vicente et al., 1994; Zhu, 1995; Ye & Zhu, 1997). There exist multiple surveys and books covering various approaches for these problems (Vicente & Calamai, 1994; Dempe, 2002; Colson et al., 2007; Bard, 2013; Sinha et al., 2017).

Ghadimi & Wang (2018) observed Eq. (1) under strong-convexity of the inner problem using the implicit function theorem, therefore asserting that the hypergradient can be computed via inverse Hessians, which requires solving a linear system at each point. Many follow up works built upon this second-order approach with additional techniques such as variance reduction, momentum, Hessian sketches, projection-free updates, or incorporating external constraints (Amini & Yousefian, 2019; Yang et al., 2021; Khanduri et al., 2021; Guo et al., 2021; Ji et al., 2021; Chen et al., 2021; Akhtar et al., 2022; Chen et al., 2022; Tsaknakis et al., 2022; Hong et al., 2023; Jiang et al., 2023; Abolfazli et al., 2023; Merchav & Sabach, 2023; Xu & Zhu, 2023; Cao et al., 2024; Dagr eou et al., 2024).

Only recently, the groundbreaking result of Liu et al. (2022) proved finite-time convergence guarantees for a fully first-order method which is based on a penalty approach. This result was soon extended to stochastic objectives (Kwon et al., 2023), with the convergence rate later improved by (Yang et al., 2023; Chen et al., 2024), and also extended to constrained bilevel problems (Yao et al., 2024; Kornowski et al., 2024). The first-order penalty paradigm also shows promise for some bilevel problems in which the inner problem is not strongly-convex (Shen & Chen, 2023; Kwon et al., 2024; Lu & Mei, 2024), which is generally a highly challenging setting (Chen et al., 2024; Bolte et al., 2024). Moreover, Pan et al. (2024) provided an efficient implementation of this paradigm, showing its effectiveness for large scale applications.

As to DP optimization, there is an extensive literature on optimization problems, both for ERM and for stochastic losses, which are either convex (Chaudhuri et al., 2011; Kifer et al., 2012; Bassily et al., 2014; Wang et al., 2017; Bassily et al., 2019; Feldman et al., 2020; Gopi et al., 2022; Carmon et al., 2023) or smooth and nonconvex (Wang et al., 2019; Tran & Cutkosky, 2022; Arora et al., 2023; Ganesh et al., 2024; Lowy et al., 2024).

To the best of our knowledge, the only existing result for DP bilevel optimization is the very recent result of Chen & Wang (2024), which differs than ours in several aspect. Their proposed algorithm is second-order, requiring evaluating Hessians, and solving corresponding linear systems at each time step, which we avoid altogether. Moreover, Chen & Wang (2024) study the *local* DP model (Kasiviswanathan et al., 2011), in which each user (i.e. ξ_i) does not reveal its individual information. Due to this more challenging setting, they can only derive guarantees for *some finite* privacy budget $\epsilon < \infty$, even as the dataset size grows $n \rightarrow \infty$. We study the common central DP model, in which a trusted curator acts on the collected data and releases a private solution, and thus are able to provide any desired privacy and accuracy guarantees with sufficiently many samples. Our work is the first to study bilevel optimization in this well-studied DP setting. We also note that Fioletto et al. (2021) studied the related problem of DP in Stackelberg games, which are certain bilevel programs which arise in game theory, aiming at designing coordination mechanisms that maintain the individual agents’ privacy.

2 PRELIMINARIES

Notation and terminology. We let bold-face letters (e.g. \mathbf{x}) denote vectors, and denote by $\mathbf{0}$ the zero vector (whenever the dimension is clear from context) and by $I_d \in \mathbb{R}^{d \times d}$ the identity matrix. $[n] := \{1, 2, \dots, n\}$. $\langle \cdot, \cdot \rangle$ denotes the standard Euclidean dot product, and $\|\cdot\|$ denotes either its induced norm for vectors or operator norm for matrices, and $\|f\|_\infty = \sup_{\mathbf{x} \in \mathcal{X}} |f(\mathbf{x})|$ denotes the sup-norm. We denote by $\text{Proj}_{\mathbb{B}(\mathbf{z}, R)}$ the projection onto the closed ball around \mathbf{z} of radius R . $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ denotes a normal (i.e., Gaussian) random variable with mean $\boldsymbol{\mu}$ and covariance Σ . We use standard big-O notation, with $\mathcal{O}(\cdot)$ hiding absolute constants (independent of problem parameters), $\tilde{\mathcal{O}}(\cdot)$ also hiding poly-logarithmic terms. We denote $f \lesssim g$ if $f = \mathcal{O}(g)$, and $f \asymp g$ if $f \lesssim g$ and $g \lesssim f$. A function $f : \mathcal{X} \subseteq \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}$ is L_0 -Lipschitz if for all $\mathbf{x}, \mathbf{y} \in \mathcal{X} : \|f(\mathbf{x}) - f(\mathbf{y})\| \leq L_0 \|\mathbf{x} - \mathbf{y}\|$; L_1 -smooth if ∇f exists and is L_1 -Lipschitz; and L_2 Hessian-smooth if $\nabla^2 f$ exists and is L_2 -Lipschitz (with respect to the operator norm). A twice-differentiable function f is μ -strongly-convex if $\nabla^2 f \succeq \mu I$, denoting by “ \succeq ” the standard PSD (“Loewner”) order on matrices.

Differential privacy. Two datasets $S, S' \in \Xi^n$ are said to be neighboring, denoted by $S \sim S'$, if they differ by only one data point. A randomized algorithm $\mathcal{A} : \Xi^n \rightarrow \mathcal{R}$ is called (ϵ, δ) differentially private (or (ϵ, δ) -DP) for $\epsilon, \delta > 0$ if for any two neighboring datasets $S \sim S'$ and measurable $E \subseteq \mathcal{R}$ in the algorithm’s range, it holds that $\Pr[\mathcal{A}(S) \in E] \leq e^\epsilon \Pr[\mathcal{A}(S') \in E] + \delta$ (Dwork et al., 2006). The basic composition property of DP states that the (possibly adaptive) composition of (ϵ_0, δ_0) -DP- and (ϵ_1, δ_1) -DP mechanisms, is $(\epsilon_0 + \epsilon_1, \delta_0 + \delta_1)$ -DP. We next recall some well known DP basics: advanced composition, the Gaussian mechanism, and privacy amplification by subsampling.

Lemma 2.1 (Advanced composition, Dwork et al., 2010). *For $\epsilon_0 < 1$, a T -fold (possibly adaptive) composition of (ϵ_0, δ_0) -DP mechanisms is (ϵ, δ) -DP for $\epsilon = \sqrt{2T \log(1/\delta_0)} \epsilon_0 + 2T \epsilon_0^2$, $\delta = (T + 1)\delta_0$.*

Lemma 2.2 (Gaussian mechanism). *Given a function $h : \Xi^b \rightarrow \mathbb{R}^d$, the Gaussian mechanism $\mathcal{M}(h) : \Xi^b \rightarrow \mathbb{R}^d$ defined as $\mathcal{M}(h)(S) := h(S) + \mathcal{N}(\mathbf{0}, \sigma^2 I_d)$ is (ϵ, δ) -DP for $\epsilon, \delta \in (0, 1)$, as long as $\sigma^2 \geq \frac{2 \log(5/4\delta)(\mathcal{S}_h)^2}{\epsilon^2}$, where $\mathcal{S}_h := \sup_{S \sim S'} \|h(S) - h(S')\|$ is the L_2 -sensitivity of h .*

Lemma 2.3 (Privacy amplification, Balle et al., 2018). *Suppose $\mathcal{M} : \Xi^b \rightarrow \mathcal{R}$ is (ϵ_0, δ_0) -DP. Then given $n \geq b$, the mechanism $\mathcal{M}' : \Xi^n \rightarrow \mathcal{R}$, $\mathcal{M}'(S) := \mathcal{M}(B)$ where $B \sim \text{Unif}(\Xi)^b$, is (ϵ, δ) -DP for $\epsilon = \log(1 + (1 - (1 - 1/n)^b)(e^{\epsilon_0} - 1))$, $\delta = \delta_0$.*

We remark that advanced composition will be used when $\epsilon_0 \lesssim \sqrt{\log(1/\delta_0)/T}$, thus the accumulated privacy scales as $\epsilon \asymp \sqrt{T} \epsilon_0$. Similarly, privacy amplification will be used when $\epsilon_0 \leq 1$, under which the privacy after subsampling scales as $\epsilon \asymp \frac{b \epsilon_0}{n}$ (since $e^{\epsilon_0} - 1 \asymp \epsilon_0$, $(1 - 1/n)^b \asymp \frac{b}{n}$ and $\log(1 + \frac{b}{n} \epsilon_0) \asymp \frac{b}{n} \epsilon_0$).

Gradient mapping. Given a point $\mathbf{x} \in \mathbb{R}^d$, and some $\mathbf{v} \in \mathbb{R}^d$, $\eta > 0$, we denote

$$\mathcal{G}_{\mathbf{v}, \eta}(\mathbf{x}) := \frac{1}{\eta} (\mathbf{x} - \mathcal{P}_{\mathbf{v}, \eta}(\mathbf{x})) , \quad \mathcal{P}_{\mathbf{v}, \eta}(\mathbf{x}) := \arg \min_{\mathbf{u} \in \mathcal{X}} \left[\langle \mathbf{v}, \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}\|^2 \right] .$$

In particular, given an L -smooth function $F : \mathbb{R}^d \rightarrow \mathbb{R}$ and $\eta \leq \frac{1}{2L}$, we denote the projected gradient (also known as reduced gradient) and the gradient (or prox) mapping, respectively, as

$$\mathcal{G}_{F, \eta}(\mathbf{x}) := \frac{1}{\eta} (\mathbf{x} - \mathcal{P}_{\nabla F, \eta}(\mathbf{x})) , \quad \mathcal{P}_{\nabla F, \eta}(\mathbf{x}) := \arg \min_{\mathbf{u} \in \mathcal{X}} \left[\langle \nabla F(\mathbf{x}), \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}\|^2 \right] .$$

The projected gradient $\mathcal{G}_{F, \eta}(\mathbf{x})$ generalizes the gradient to the possibly constrained setting: for points $\mathbf{x} \in \mathcal{X}$ sufficiently far from the boundary of \mathcal{X} , $\mathcal{G}_{F, \eta}(\mathbf{x}) = \nabla F(\mathbf{x})$, namely it simply reduces to the gradient. See the textbooks (Nesterov, 2013; Lan, 2020) for additional details. We will recall a useful fact, which asserts that the mapping $\mathcal{G}_{\mathbf{v}, \eta}(\mathbf{x})$ is non-expansive with respect to \mathbf{v} :

Lemma 2.4. *For any $\mathbf{x}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^d$, $\eta > 0$: $\|\mathcal{G}_{\mathbf{v}, \eta}(\mathbf{x}) - \mathcal{G}_{\mathbf{w}, \eta}(\mathbf{x})\| \leq \|\mathbf{v} - \mathbf{w}\|$.*

Lemma 2.4 is important in our analysis, since as we will argue later (in Section 3.1), gradient estimates must be *inexact* in the bilevel setting to satisfy privacy, and Lemma 2.4 will allow us to control the error due to this inexactness. Although this result is known (cf. Ghadimi et al. 2016), we reprove it in Appendix C for completeness.

2.1 SETTING

We impose the following assumptions, all of which are standard in the bilevel optimization literature.

Assumption 2.5. For (BO) with either (ERM) or (Pop), we assume the following hold:

- i. $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ is a closed convex set.
- ii. $F(\mathbf{x}_0) - \inf_{\mathbf{x} \in \mathcal{X}} F(\mathbf{x}) \leq \Delta_F$ for some initial point $\mathbf{x}_0 \in \mathcal{X}$.
- iii. f is twice differentiable, and L_1^f -smooth.
- iv. For all $\xi \in \Xi$: $f(\cdot, \cdot; \xi)$ is L_0^f -Lipschitz (hence, so is f).
- v. g is L_2^g -Hessian-smooth, and for all $\mathbf{x} \in \mathcal{X}$: $g(\mathbf{x}, \cdot)$ is μ_g -strongly-convex.
- vi. For all $\xi \in \Xi$: $g(\cdot, \cdot; \xi)$ is L_1^g -smooth (hence, so is g).

As mentioned, these assumptions are standard in the study of bilevel optimization problems and are shared by nearly all of the previous works we discussed. In particular, the strong convexity of $g(\mathbf{x}, \cdot)$ ensures that $\mathbf{y}^*(\mathbf{x})$ is uniquely defined, which is generally required in establishing the regularity of the hyperobjective. Indeed, it is known that dropping this assumption, can, in general, lead to pathological behaviors not amenable for algorithmic guarantees (cf. Chen et al. 2024; Bolte et al. 2024 and discussions therein). For the purpose of differential privacy though, the strong convexity of $g(\mathbf{x}, \cdot)$ raises a subtle issue. As the standard assumption in the DP optimization literature is that the component functions are Lipschitz, which allows privatization of gradients using sensitivity arguments, strongly-convex objectives cannot be Lipschitz over the entire Euclidean space.³ Therefore, strongly-convex objectives are regularly analyzed in the DP setting under the additional assumption that the domain of interest is bounded. For bilevel problems, the domain of interest for \mathbf{y} is the lower level solution set, thus we impose the following assumption.

Assumption 2.6. There exists a compact set $\mathcal{Y} \subset \mathbb{R}^{d_y}$ with $\{\mathbf{y}^*(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} \subseteq \mathcal{Y}$, such that for all $\mathbf{x} \in \mathcal{X}$, $\xi \in \Xi$: $g(\mathbf{x}, \cdot; \xi)$ is L_0^g -Lipschitz over \mathcal{Y} .

Remark 2.7. Note that $\text{diam}(\mathcal{Y}) \leq 2L_0^g/\mu_g$. Indeed, fixing some $\mathbf{x} \in \mathcal{X}$, since $g(\mathbf{x}, \cdot; \xi)$ is L_0^g -Lipschitz over \mathcal{Y} for all $\xi \in \Xi$, then so is $g(\mathbf{x}, \cdot)$. Moreover, by μ_g -strong-convexity, we get that for all $\mathbf{y} \in \mathcal{Y}$: $\mu_g \|\mathbf{y} - \mathbf{y}^*(\mathbf{x})\| \leq \|\nabla_{\mathbf{y}} g(\mathbf{x}, \mathbf{y})\| \leq L_0^g$. Hence $\mathcal{Y} \subseteq \mathbb{B}(\mathbf{y}^*(\mathbf{x}), L_0^g/\mu_g)$, which is of diameter $2L_0^g/\mu_g$.

Following Assumptions 2.5 and 2.6, we denote $\ell := \max\{L_0^f, L_1^f, L_0^g, L_1^g, L_2^g\}$, $\kappa := \ell/\mu_g$.

3 ALGORITHM FOR BILEVEL ERM

In this section, we consider the ERM bilevel problem, namely (BO) with (ERM), for which we denote the hyperobjective by F_S . Our algorithm is presented in Algorithm 1. We prove the following result:

Theorem 3.1. Assume 2.5 and 2.6 hold, and that $\alpha \leq \ell \kappa^3 \min\{\frac{1}{2\kappa}, \frac{L_0^g}{L_1^f}, \frac{L_1^g}{L_1^f}, \frac{\Delta_F}{\ell \kappa}\}$. Then there is an assignment of parameters $\lambda \asymp \ell \kappa^3 \alpha^{-1}$, $\sigma^2 \asymp \ell^2 \kappa^2 T \log(T/\delta) \epsilon^{-2} n^{-2}$, $\eta \asymp \ell^{-1} \kappa^{-3}$, $T \asymp \Delta_F \ell \kappa^3 \alpha^{-2}$, such that running Algorithm 1 satisfies (ϵ, δ) -DP, and returns \mathbf{x}_{out} such that with probability at least $1 - \gamma$:

$$\|\mathcal{G}_{F_S, \eta}(\mathbf{x}_{\text{out}})\| \leq \alpha = \tilde{\mathcal{O}} \left(K_1 \left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + K_2 \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} \right),$$

where $K_1 = \mathcal{O}(\Delta_F^{1/4} \ell^{3/4} \kappa^{5/4})$, $K_2 = \mathcal{O}(\Delta_F^{1/6} \ell^{1/2} \kappa^{11/6})$.

Remark 3.2. Recall that when $\mathcal{X} = \mathbb{R}^{d_x}$, then $\mathcal{G}_{F_S, \eta}(\mathbf{x}_{\text{out}}) = \nabla F_S(\mathbf{x}_{\text{out}})$.

³If $g(\mathbf{x}, \cdot; \xi)$ were Lipschitz over \mathbb{R}^{d_y} for all $\xi \in \Xi$, then so would $g(\mathbf{x}, \cdot)$, contradicting strong convexity.

Algorithm 1 DP Bilevel

-
- 1: **Input:** Initialization $(\mathbf{x}_0, \mathbf{y}_0) \in \mathcal{X} \times \mathcal{Y}$, privacy budget (ϵ, δ) , penalty $\lambda > 0$, noise level $\sigma^2 > 0$, step size $\eta > 0$, iteration budget $T \in \mathbb{N}$.
 - 2: **for** $t = 0, \dots, T - 1$ **do**
 - 3: Apply $(\frac{\epsilon}{\sqrt{18T}}, \frac{\delta}{3T})$ -DP-Loc-GD (Algorithm 2) to solve ▷ Strongly-convex problems
 - $$\tilde{\mathbf{y}}_t \approx \arg \min_{\mathbf{y}} g(\mathbf{x}_t, \mathbf{y})$$

$$\tilde{\mathbf{y}}_t^\lambda \approx \arg \min_{\mathbf{y}} [f(\mathbf{x}_t, \mathbf{y}) + \lambda \cdot g(\mathbf{x}_t, \mathbf{y})]$$
 - 4: $\tilde{\mathbf{g}}_t = \nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda (\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t)) + \nu_t$, where $\nu_t \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_{d_x})$
 - 5: $\mathbf{x}_{t+1} = \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \langle \tilde{\mathbf{g}}_t, \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}_t\|^2 \right\}$ ▷ If $\mathcal{X} = \mathbb{R}^{d_x}$, then $\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \tilde{\mathbf{g}}_t$
 - 6: **end for**
 - 7: $t_{\text{out}} = \arg \min_{t \in \{0, \dots, T-1\}} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|$.
 - 8: **Output:** $\mathbf{x}_{t_{\text{out}}}$.
-

Algorithm 2 DP-Loc-GD

-
- 1: **Input:** Objective $h : \mathbb{R}^{d_y} \rightarrow \mathbb{R}$, initialization $\mathbf{y}_0 \in \mathcal{Y}$, privacy budget (ϵ', δ') , number of rounds $M \in \mathbb{N}$, noise level $\sigma_{\text{GD}}^2 > 0$, step sizes $(\eta_t)_{t=0}^{T-1}$, iteration budget $T_{\text{GD}} \in \mathbb{N}$, radii $(R_m)_{m=0}^{M-1} > 0$.
 - 2: $\mathbf{y}_0^0 = \mathbf{y}_0$
 - 3: **for** $m = 0, \dots, M - 1$ **do**
 - 4: **for** $t = 0, \dots, T_{\text{GD}} - 1$ **do**
 - 5: $\mathbf{y}_{t+1}^m = \text{Proj}_{\mathbb{B}(\mathbf{y}_0^m, R_m)} [\mathbf{y}_t^m - \eta_t (\nabla h(\mathbf{y}_t^m) + \nu_t)]$, where $\nu_t^m \sim \mathcal{N}(\mathbf{0}, \sigma_{\text{GD}}^2 I_{d_y})$
 - 6: **end for**
 - 7: $\mathbf{y}_0^{m+1} = \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{y}_t^m$
 - 8: **end for**
 - 9: **Output:** $\mathbf{y}_{\text{out}} = \mathbf{y}_0^M$.
-

3.1 ANALYSIS OVERVIEW

In this section, we will go over the main ideas that appear in the proof of Theorem 3.1, all which are provided in full detail in Appendix A. We start by introducing some useful notation: Given $\lambda > 0$, we denote the penalty function

$$\mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}) := f(\mathbf{x}, \mathbf{y}) + \lambda [g(\mathbf{x}, \mathbf{y}) - g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))],$$

and further denote

$$\mathcal{L}_\lambda^*(\mathbf{x}) := \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})), \quad \mathbf{y}^\lambda(\mathbf{x}) := \arg \min_{\mathbf{y}} \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}).$$

The starting point of our analysis is the following result, underlying the previously discussed recent advancements in (non-private) first-order bilevel optimization:

Lemma 3.3 (Kwon et al. 2023; 2024; Chen et al. 2024). *For $\lambda \geq 2L_1^f/\mu_g$, the following hold:*

- a. $\|\mathcal{L}_\lambda^* - F\|_\infty = \mathcal{O}(\ell\kappa/\lambda)$.
- b. $\|\nabla \mathcal{L}_\lambda^* - \nabla F\|_\infty = \mathcal{O}(\ell\kappa^3/\lambda)$.
- c. \mathcal{L}_λ^* is $\mathcal{O}(\ell\kappa^3)$ -smooth (independently of λ).

In other words, the lemma shows that for sufficiently large penalty λ , \mathcal{L}_λ^* is a smooth approximation of the hyperobjective F , and that it suffices to minimize the gradient norm of \mathcal{L}_λ^* in order to get a hypergradient guarantee in terms of ∇F . Moreover, note that $\nabla \mathcal{L}_\lambda^*$ can be computed entirely in a

first-order fashion, since by construction $\mathcal{L}_\lambda^*(\mathbf{x}) = \arg \min_{\mathbf{y}} \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y})$, and therefore it holds that

$$\begin{aligned} \nabla \mathcal{L}_\lambda^*(\mathbf{x}) &= \nabla_x \mathcal{L}_\lambda^*(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \nabla_x \mathbf{y}^\lambda(\mathbf{x})^\top \underbrace{\nabla_y \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}))}_{=0} \\ &= \nabla_x f(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \lambda (\nabla_x g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \nabla_x g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))) . \end{aligned} \quad (3)$$

This observation raises a subtle privacy issue: Since $\mathbf{y}^*(\mathbf{x}), \mathbf{y}^\lambda(\mathbf{x})$ are required in order to compute the gradient $\nabla \mathcal{L}_\lambda^*(\mathbf{x})$, and are defined as the minimizers of $g(\mathbf{x}, \cdot), \mathcal{L}_\lambda(\mathbf{x}, \cdot)$ which are data-dependent, we cannot simply compute them up to arbitrarily small accuracy under the DP constraint. In other words, even deciding *where* to invoke the gradient oracles, can already leak user information, hence breaking privacy before the gradients are even computed. We therefore must resort to approximating them using an auxiliary private method, for which we use DP-Loc(alized)-GD (Algorithm 2).⁴ We then crucially rely on the fact that $g(\mathbf{x}, \cdot), \mathcal{L}_\lambda(\mathbf{x}, \cdot)$ are both strongly-convex, which implies that optimizing them produces $\tilde{\mathbf{y}}_t, \tilde{\mathbf{y}}_t^\lambda$ such that the *distances* to the minimizers, namely $\|\tilde{\mathbf{y}}_t - \mathbf{y}^*(\mathbf{x}_t)\|, \|\tilde{\mathbf{y}}_t^\lambda - \mathbf{y}^\lambda(\mathbf{x}_t)\|$, are small. The distance bound is key, as Eq. (3) allows using the smoothness of f, g to translate the distance bounds into an inexact (i.e. biased) gradient oracle for $\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t)$, computed at the private points $\tilde{\mathbf{y}}_t, \tilde{\mathbf{y}}_t^\lambda$. Using this analysis we obtain the following guarantee:

Lemma 3.4. *If $\lambda \geq \max\{\frac{2L_1^g}{\mu_g}, \frac{L_0^f}{L_0^g}, \frac{L_1^f}{L_1^g}\}$, then there is $\beta = \tilde{\mathcal{O}}\left(\frac{\lambda \ell \kappa \sqrt{d_y T}}{\epsilon n}\right)$ such that with probability at least $1 - \gamma$, for all $t \in \{0, \dots, T - 1\}$:*

$$\|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - [\nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda (\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))]\| \leq \beta .$$

Having constructed an inexact gradient oracle, we can privatize its response using the standard Gaussian mechanism. Recalling that the noise variance required to ensure privacy is tied to the component Lipschitz constants, we note that $\mathcal{L}_\lambda^*(\mathbf{x})$ decomposes as the finite-sum $\mathcal{L}_\lambda^*(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\lambda,i}^*(\mathbf{x})$, where

$$\mathcal{L}_{\lambda,i}^*(\mathbf{x}) := f(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}); \xi_i) + \lambda [g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}); \xi_i) - g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}); \xi_i)] .$$

At first glance, a naive application of the chain rule and the triangle inequality would bound the Lipschitz constant of $\mathcal{L}_{\lambda,i}^*$ by approximately $\text{Lip}(\mathbf{y}^*)(L_0^f + \lambda L_0^g) \lesssim \lambda \text{Lip}(\mathbf{y}^*) L_0^g$, where $\text{Lip}(\mathbf{y}^*)$ is the Lipschitz constant of $\mathbf{y}^*(\mathbf{x}) : \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_y}$. Unfortunately, this bound grows with the penalty parameter λ , which will eventually be set large, and in particular, will grow with the dataset size n . We therefore derive the following lemma, showing that applying a more nuanced analysis allows obtaining a significantly tighter Lipschitz bound, independent of λ :

Lemma 3.5. $\mathcal{L}_{\lambda,i}^*$ is $\mathcal{O}(\ell \kappa)$ -Lipschitz.

Finally, having constructed a private inexact stochastic oracle response for the smooth approximation \mathcal{L}_λ^* , we analyze an outer loop (Line 5 of Algorithm 1), showing that is provably robust to inexact and noisy gradients. We then employ a stopping criteria which makes use of the already-privatized iterates, thus avoiding the need of additional noise in choosing the smallest gradient. In particular, we show that the corresponding process gets to a point with small (projected-)gradient norm, as stated below:

Proposition 3.6. *Suppose $h : \mathbb{R}^d \rightarrow \mathbb{R}$ is L -smooth, that $\|\tilde{\nabla} h(\cdot) - \nabla h(\cdot)\| \leq \beta$, and consider the following update rule with $\eta = \frac{1}{2L}$:*

$$\mathbf{x}_{t+1} = \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \left\langle \tilde{\nabla} h(\mathbf{x}_t) + \nu_t, \mathbf{u} \right\rangle + \frac{1}{2\eta} \|\mathbf{x}_t - \mathbf{u}\|^2 \right\}, \quad \nu_t \sim \mathcal{N}(0, \sigma^2 I),$$

with the output rule $\mathbf{x}_{\text{out}} := \mathbf{x}_{t_{\text{out}}}$, $t_{\text{out}} := \arg \min_{t \in \{0, \dots, T-1\}} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|$. If $\alpha > 0$ is such that $\alpha \geq C \max\{\beta, \sigma \sqrt{d \log(T/\gamma)}\}$ for a sufficiently large absolute constant $C > 0$, then with probability at least $1 - \gamma$: $\|\mathcal{G}_{h,\eta}(\mathbf{x}_{\text{out}})\| \leq \alpha$ for $T = \mathcal{O}\left(\frac{L(h(\mathbf{x}_0) - \inf h)}{\alpha^2}\right)$.

Overall, applying Proposition 3.6 to $h = \mathcal{L}_\lambda^*$, we see that the (projected-)gradient norm can be as small as $\max\{\beta, \sigma \sqrt{d_x}\}$, up to logarithmic terms. Accounting for the smallest possible inexactness β and noise addition σ that ensure the the inner and outer loops, respectively, are both sufficiently private, we conclude the proof of Theorem 3.1; the full details appear in Appendix A.

⁴We can replace the inner solver by any DP method that guarantees with high probability the optimal rate for strongly-convex objectives, as we further discuss in Appendix B.

432 $\lim_{n \rightarrow \infty} \|\mathcal{G}(\mathbf{x}_{\text{out}})\| = 0$). It is interesting to note that the additional $1/b_{\text{out}}$ term shows up in the
 433 analysis only as an upper bound on the sub-Gaussian norm of the mini-batch gradient estimator.
 434 Thus, in applications for which some (possibly constant) batch size results in reasonably accurate
 435 gradients, the result above should hold with the outer mini-batch gradient’s standard deviation re-
 436 placing $1/b_{\text{out}}$, which is to be expected anyhow for high probability guarantees.

437 The difference between Algorithm 3 and Algorithm 1, is that both the inner and outer loops sample
 438 mini-batch gradients. The inner loop guarantee is the same regardless of the inner-batch size b_{in} ,
 439 since for strongly-convex objectives it is possible to prove the same convergence rate guarantee
 440 for DP optimization in any case (as further discussed in Appendix B). As to the outer loop (Line
 441 5), we apply standard concentration bounds to argue about the quality of the gradient estimates —
 442 hence the additive $1/b_{\text{out}}$ factor — and rely on our analysis of the outer loop with inexact gradients
 443 (which are now even less exact due to sampling stochasticity). We remark that compared to the
 444 classic analysis of Ghadimi et al. (2016) for constrained nonconvex optimization, we derive high
 445 probability bounds without requiring several re-runs of the algorithm. We further remark that we
 446 analyze mini-batch sampling with replacement for simplicity, though the same guarantees (up to
 447 constants) can be derived for sampling at each time step without replacement, at the cost of a more
 448 involved analysis.

450 5 GENERALIZING FROM ERM TO POPULATION LOSS

451 In this section, we move to consider stochastic (population) objectives, the problem (BO) with (Pop).
 452 We denote the population hyperobjective by $F_{\mathcal{P}}$, and as before F_S denotes the empirical objective,
 453 where $S \sim \mathcal{P}^n$. We prove the following result:

454 **Theorem 5.1.** Under Assumptions 2.5 and 2.6, if the preconditions of Theorem 3.1 hold, then Algo-
 455 rithm 1 is (ϵ, δ) -DP, and returns \mathbf{x}_{out} such that with probability at least $1 - \gamma$:

$$456 \|\mathcal{G}_{F_{\mathcal{P}}, \eta}(\mathbf{x}_{\text{out}})\| \leq \alpha = \tilde{\mathcal{O}} \left(K_1 \left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + K_2 \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} + K_3 \left(\frac{d_x}{n} \right)^{1/2} \right),$$

457 where $K_1 = \mathcal{O}(\Delta_F^{1/4} \ell^{1/4} \kappa^{5/4})$, $K_2 = \mathcal{O}(\Delta_F^{1/6} \ell^{1/2} \kappa^{11/6})$, $K_3 = \mathcal{O}(\ell \kappa)$. Similarly, if the pre-
 458 conditions of Theorem 4.1 hold, then for any batch sizes $b_{\text{in}}, b_{\text{out}} \in [n]$, Algorithm 3 is (ϵ, δ) -DP, and
 459 returns \mathbf{x}_{out} such that with probability at least $1 - \gamma$:

$$460 \|\mathcal{G}_{F_{\mathcal{P}}, \eta}(\mathbf{x}_{\text{out}})\| \leq \alpha = \tilde{\mathcal{O}} \left(K_1 \left(\frac{\sqrt{d_x}}{\epsilon n} \right)^{1/2} + K_2 \left(\frac{\sqrt{d_y}}{\epsilon n} \right)^{1/3} + K_3 \cdot \frac{1}{b_{\text{out}}} + K_3 \left(\frac{d_x}{n} \right)^{1/2} \right).$$

461 The proof of Theorem 5.1 relies on arguing that the hyperobjective is Lipschitz, and applying a
 462 uniform convergence bound for bounded gradients, which further implies uniform convergence of
 463 projected gradients by Lemma 2.4.

464 6 DISCUSSION

465 In this paper, we studied differentially-private bilevel optimization, and proposed the first algorithms
 466 to solve this problem that enable any desired privacy guarantee, while also requiring only gradient
 467 queries. Our provided guarantees hold both for constrained and unconstrained settings, cover em-
 468 pirical and population losses alike, and account for mini-batched gradients.

469 Our work leaves open several directions for future research. First, it is likely that the rate derived
 470 in this work can be improved. Specifically, for “single”-level DP nonconvex optimization, Arora
 471 et al. (2023) showed that incorporating variance reduction leads to gradient bounds that decay faster
 472 with the sample size. Applying this for DP bilevel optimization as the outer loop would require,
 473 according to our analysis, to evaluate the cost of inexact gradients in variance-reduced methods,
 474 which we leave for future work.

475 Another open direction is understanding whether mini-batch algorithms can avoid the additive
 476 $1/b_{\text{out}}$ factor in the *unconstrained* case $\mathcal{X} = \mathbb{R}^{d_x}$. As previously discussed, for constrained prob-
 477 lems, even single-level nonconvex algorithms suffer from this batch dependence (Ghadimi et al.,

2016). Nonetheless, for unconstrained problems, Ghadimi & Lan (2013) showed that setting a smaller stepsize, roughly on the order of α^2/σ^2 , converges to a point with gradient bounded by α after $\mathcal{O}(\alpha^{-4})$ steps, even for $b_{\text{out}} = 1$. Applying this to DP bilevel unconstrained optimization, would require analyzing SGD under biased gradients, and accounting for the larger privacy loss due to the slower convergence rate (compared to $\mathcal{O}(\alpha^{-2})$ in our case), both of which seem feasible.

Lastly, an important direction is of course empirical validation of our proposed methods. At a high level, our methods are privatized variants of the first-order penalty approach for bilevel optimization, which has been substantially scaled up following initial theoretically-focused works, confirming this paradigm as highly effective in some large scale non-private applications (Pan et al., 2024). While our analysis provides conservative (worst-case) estimates for the convergence rate under privatization of both the upper and lower level problems, it would be interesting to explore the actual cost of privatization seen in practice for these problems. As this work is a theoretically focused, we leave this for future research.

REFERENCES

- Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.
- Nazanin Abolfazli, Ruichen Jiang, Aryan Mokhtari, and Erfan Yazdandoost Hamedani. An inexact conditional gradient method for constrained bilevel optimization. *arXiv preprint arXiv:2306.02429*, 2023.
- Zeeshan Akhtar, Amrit Singh Bedi, Srujan Teja Thomdapu, and Ketan Rajawat. Projection-free stochastic bi-level optimization. *IEEE Transactions on Signal Processing*, 70:6332–6347, 2022.
- Mostafa Amini and Farzad Yousefian. An iterative regularized incremental projected subgradient method for a class of bilevel optimization problems. In *2019 American Control Conference (ACC)*, pp. 4069–4074. IEEE, 2019.
- G Anandalingam and DJ White. A solution method for the linear static stackelberg problem using penalty functions. *IEEE Transactions on automatic control*, 35(10):1170–1173, 1990.
- Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016.
- Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- Raman Arora, Raef Bassily, Tomás González, Cristóbal A Guzmán, Michael Menart, and Enayat Ullah. Faster rates of convergence to stationary points in differentially private optimization. In *International Conference on Machine Learning*, pp. 1060–1092. PMLR, 2023.
- Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in neural information processing systems*, 31, 2018.
- Jonathan F Bard. *Practical bilevel optimization: algorithms and applications*, volume 30. Springer Science & Business Media, 2013.
- Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th annual symposium on foundations of computer science*, pp. 464–473. IEEE, 2014.
- Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Guha Thakurta. Private stochastic convex optimization with optimal rates. *Advances in neural information processing systems*, 32, 2019.
- Yoshua Bengio. Gradient-based optimization of hyperparameters. *Neural computation*, 12(8):1889–1900, 2000.

- 540 Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differ-
541 entiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.
542
- 543 Jérôme Bolte, Quoc-Tung Le, Edouard Pauwels, and Samuel Vaïter. Geometric and computational
544 hardness of bilevel programming. *arXiv preprint arXiv:2407.12372*, 2024.
- 545 Jerome Bracken and James T McGill. Mathematical programs with optimization problems in the
546 constraints. *Operations research*, 21(1):37–44, 1973.
547
- 548 Jincheng Cao, Ruichen Jiang, Nazanin Abolfazli, Erfan Yazdandoost Hamedani, and Aryan
549 Mokhtari. Projection-free methods for stochastic simple bilevel optimization with convex lower-
550 level problem. *Advances in Neural Information Processing Systems*, 36, 2024.
- 551 Yair Carmon, Arun Jambulapati, Yujia Jin, Yin Tat Lee, Daogao Liu, Aaron Sidford, and Kevin
552 Tian. Resqueing parallel and private stochastic convex optimization. In *2023 IEEE 64th Annual
553 Symposium on Foundations of Computer Science (FOCS)*, pp. 2031–2058. IEEE, 2023.
554
- 555 Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical
556 risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.
557
- 558 Lesi Chen, Jing Xu, and Jingzhao Zhang. On finding small hyper-gradients in bilevel optimization:
559 Hardness results and improved analysis. In *Proceedings of Thirty Seventh Conference on Learning
560 Theory*, volume 247 of *Proceedings of Machine Learning Research*, pp. 947–980. PMLR, 2024.
- 561 Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic
562 gradient methods for bilevel problems. *Advances in Neural Information Processing Systems*, 34:
563 25294–25307, 2021.
564
- 565 Tianyi Chen, Yuejiao Sun, Quan Xiao, and Wotao Yin. A single-timescale method for stochastic
566 bilevel optimization. In *International Conference on Artificial Intelligence and Statistics*, pp.
567 2466–2488. PMLR, 2022.
- 568 Ziqin Chen and Yongqiang Wang. Locally differentially private decentralized stochastic bilevel
569 optimization with guaranteed convergence accuracy. In *Forty-first International Conference on
570 Machine Learning*, 2024.
571
- 572 Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of
573 operations research*, 153:235–256, 2007.
- 574 Mathieu Dagréou, Thomas Moreau, Samuel Vaïter, and Pierre Ablin. A lower bound and a near-
575 optimal algorithm for bilevel empirical risk minimization. In *International Conference on Artifi-
576 cial Intelligence and Statistics*, pp. 82–90. PMLR, 2024.
577
- 578 Stephan Dempe. *Foundations of bilevel programming*. Springer Science & Business Media, 2002.
579
- 580 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity
581 in private data analysis. In *Third Theory of Cryptography Conference*, pp. 265–284. Springer,
582 2006.
- 583 Cynthia Dwork, Guy N Rothblum, and Salil Vadhan. Boosting and differential privacy. In *2010
584 IEEE 51st annual symposium on foundations of computer science*, pp. 51–60. IEEE, 2010.
585
- 586 Simin Fan, Matteo Pagliardini, and Martin Jaggi. Doge: Domain reweighting with generalization
587 estimation. In *Forty-first International Conference on Machine Learning*, 2024.
- 588 Vitaly Feldman, Tomer Koren, and Kunal Talwar. Private stochastic convex optimization: optimal
589 rates in linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of
590 Computing*, pp. 439–449, 2020.
591
- 592 Ferdinando Fioretto, Lesia Mitridati, and Pascal Van Hentenryck. Differential privacy for stackel-
593 berg games. In *Proceedings of the Twenty-Ninth International Conference on International Joint
Conferences on Artificial Intelligence*, pp. 3480–3486, 2021.

- 594 Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse
595 gradient-based hyperparameter optimization. In *International Conference on Machine Learning*,
596 pp. 1165–1173. PMLR, 2017.
- 597 Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazi, and Massimiliano Pontil. Bilevel
598 programming for hyperparameter optimization and meta-learning. In *International conference on*
599 *machine learning*, pp. 1568–1577. PMLR, 2018.
- 600
- 601 Arun Ganesh, Daogao Liu, Sewoong Oh, and Abhradeep Guha Thakurta. Private (stochastic) non-
602 convex optimization revisited: Second-order stationary points and excess risks. *Advances in*
603 *Neural Information Processing Systems*, 36, 2024.
- 604 Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochas-
605 tic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.
- 606
- 607 Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint*
608 *arXiv:1802.02246*, 2018.
- 609 Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Mini-batch stochastic approximation meth-
610 ods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1):267–
611 305, 2016.
- 612 Sivakanth Gopi, Yin Tat Lee, and Daogao Liu. Private convex optimization via exponential mecha-
613 nism. In *Conference on Learning Theory*, pp. 1948–1989. PMLR, 2022.
- 614
- 615 David Grangier, Pierre Ablin, and Awni Hannun. Bilevel optimization to learn training distributions
616 for language modeling under domain shift. In *NeurIPS 2023 Workshop on Distribution Shifts:*
617 *New Frontiers with Foundation Models*, 2023.
- 618 Zhishuai Guo, Quanqi Hu, Lijun Zhang, and Tianbao Yang. Randomized stochastic variance-
619 reduced methods for multi-task stochastic bilevel optimization. *arXiv preprint arXiv:2105.02266*,
620 2021.
- 621
- 622 Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale stochastic algorithm
623 framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM*
624 *Journal on Optimization*, 33(1):147–180, 2023.
- 625 Yo Ishizuka and Eitaro Aiyoshi. Double penalty method for bilevel optimization problems. *Annals*
626 *of Operations Research*, 34(1):73–88, 1992.
- 627
- 628 Kaiyi Ji, Jason D Lee, Yingbin Liang, and H Vincent Poor. Convergence of meta-learning with task-
629 specific adaptation over partial parameters. *Advances in Neural Information Processing Systems*,
630 33:11490–11500, 2020.
- 631 Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced
632 design. In *International conference on machine learning*, pp. 4882–4892. PMLR, 2021.
- 633 Ruichen Jiang, Nazanin Abolfazli, Aryan Mokhtari, and Erfan Yazdandoost Hamedani. A condi-
634 tional gradient-based method for simple bilevel optimization with convex lower-level problem.
635 In *International Conference on Artificial Intelligence and Statistics*, pp. 10305–10323. PMLR,
636 2023.
- 637
- 638 Yibo Jiang and Victor Veitch. Invariant and transportable representations for anti-causal domain
639 shifts. *Advances in Neural Information Processing Systems*, 35:20782–20794, 2022.
- 640 Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan. A short
641 note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint*
642 *arXiv:1902.03736*, 2019.
- 643
- 644 Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam
645 Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.
- 646 Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang.
647 A near-optimal algorithm for stochastic bilevel optimization via double-momentum. *Advances in*
neural information processing systems, 34:30271–30283, 2021.

- 648 Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization
649 and high-dimensional regression. In *Conference on Learning Theory*, pp. 25–1. JMLR Workshop
650 and Conference Proceedings, 2012.
- 651 Guy Kornowski, Swati Padmanabhan, Kai Wang, Zhe Zhang, and Suvrit Sra. First-order methods
652 for linearly constrained bilevel optimization. *arXiv preprint arXiv:2406.12771*, 2024.
- 653 Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method
654 for stochastic bilevel optimization. In *International Conference on Machine Learning*, pp. 18083–
655 18113. PMLR, 2023.
- 656 Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. On penalty methods for
657 nonconvex bilevel optimization and first-order stochastic approximation. In *The Twelfth Interna-
658 tional Conference on Learning Representations*, 2024.
- 659 Guanghui Lan. *First-order and stochastic optimization methods for machine learning*, volume 1.
660 Springer, 2020.
- 661 Bo Liu, Mao Ye, Stephen Wright, Peter Stone, and Qiang Liu. Bome! bilevel optimization made
662 easy: A simple first-order approach. *Advances in neural information processing systems*, 35:
663 17248–17262, 2022.
- 664 Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv
665 preprint arXiv:1806.09055*, 2018.
- 666 Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by
667 implicit differentiation. In *International conference on artificial intelligence and statistics*, pp.
668 1540–1552. PMLR, 2020.
- 669 Andrew Lowy, Jonathan Ullman, and Stephen Wright. How to make the gradients small privately:
670 Improved rates for differentially private non-convex optimization. In *Forty-first International
671 Conference on Machine Learning*, 2024.
- 672 Zhaosong Lu and Sanyou Mei. First-order penalty methods for bilevel optimization. *SIAM Journal
673 on Optimization*, 34(2):1937–1969, 2024.
- 674 Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimiza-
675 tion through reversible learning. In *International conference on machine learning*, pp. 2113–2122.
676 PMLR, 2015.
- 677 Song Mei, Yu Bai, and Andrea Montanari. The landscape of empirical risk for nonconvex losses.
678 *The Annals of Statistics*, 46(6A):2747–2774, 2018.
- 679 Roey Merchav and Shoham Sabach. Convex bi-level optimization problems with nonsmooth outer
680 objective function. *SIAM Journal on Optimization*, 33(4):3114–3142, 2023.
- 681 Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer
682 Science & Business Media, 2013.
- 683 Rui Pan, Jipeng Zhang, Xingyuan Pan, Renjie Pi, Xiaoyu Wang, and Tong Zhang. Scalebio: Scalable
684 bilevel optimization for llm data reweighting. *arXiv preprint arXiv:2406.19976*, 2024.
- 685 Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with im-
686 plicit gradients. *Advances in neural information processing systems*, 32, 2019.
- 687 Ohad Shamir. A variant of azuma’s inequality for martingales with subgaussian tails. *arXiv preprint
688 arXiv:1110.2392*, 2011.
- 689 Han Shen and Tianyi Chen. On penalty-based bilevel gradient descent method. In *International
690 Conference on Machine Learning*, pp. 30992–31015. PMLR, 2023.
- 691 Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical
692 to evolutionary approaches and applications. *IEEE transactions on evolutionary computation*, 22
693 (2):276–295, 2017.

- 702 Hoang Tran and Ashok Cutkosky. Momentum aggregation for private non-convex erm. *Advances*
703 *in Neural Information Processing Systems*, 35:10996–11008, 2022.
704
- 705 Ioannis Tsaknakis, Prashant Khanduri, and Mingyi Hong. An implicit gradient-type method for
706 linearly constrained bilevel problems. In *ICASSP 2022-2022 IEEE International Conference on*
707 *Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5438–5442. IEEE, 2022.
- 708 Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*,
709 volume 47. Cambridge university press, 2018.
710
- 711 Luis Vicente, Gilles Savard, and Joaquim Júdice. Descent approaches for quadratic bilevel program-
712 ming. *Journal of Optimization theory and applications*, 81(2):379–399, 1994.
- 713 Luis N Vicente and Paul H Calamai. Bilevel and multilevel programming: A bibliography review.
714 *Journal of Global optimization*, 5(3):291–306, 1994.
715
- 716 Di Wang, Minwei Ye, and Jinhui Xu. Differentially private empirical risk minimization revisited:
717 Faster and more general. *Advances in Neural Information Processing Systems*, 30, 2017.
- 718 Di Wang, Changyou Chen, and Jinhui Xu. Differentially private empirical risk minimization with
719 non-convex loss functions. In *International Conference on Machine Learning*, pp. 6526–6535.
720 PMLR, 2019.
721
- 722 Douglas J White and G Anandalingam. A penalty function approach for solving bi-level linear
723 programs. *Journal of Global Optimization*, 3:397–419, 1993.
- 724 Siyuan Xu and Minghui Zhu. Efficient gradient approximation method for constrained bilevel opt-
725 imization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp.
726 12509–12517, 2023.
727
- 728 Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. *Ad-*
729 *vances in Neural Information Processing Systems*, 34:13670–13682, 2021.
- 730 Yifan Yang, Peiyao Xiao, and Kaiyi Ji. Achieving $\mathcal{O}(\epsilon^{-1.5})$ complexity in hessian/jacobian-free
731 stochastic bilevel optimization. In *Proceedings of the 37th International Conference on Neural*
732 *Information Processing Systems*, pp. 39491–39503, 2023.
- 733 Wei Yao, Chengming Yu, Shangzhi Zeng, and Jin Zhang. Constrained bi-level optimization: Proxi-
734 mal lagrangian value function approach and hessian-free algorithm, 2024.
735
- 736 JJ Ye and DL Zhu. Exact penalization and necessary optimality conditions for generalized bilevel
737 programming problems. *SIAM Journal on optimization*, 7(2):481–507, 1997.
738
- 739 DL Zhu. Optimality conditions for bilevel programming problems. *Optimization*, 33(1):9–27, 1995.
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755

A PROOFS

Throughout the proof section, we abbreviate $f_i(\cdot) = f(\cdot; \xi_i)$, $g_i(\cdot) = g(\cdot; \xi_i)$, $F = F_S$.

A.1 PROOF OF LEMMA 3.4

Note that the two sub-problems solved by Line 3 of Algorithm 1 are strongly-convex and admit Lipschitz components over \mathcal{Y} ; $g(\mathbf{x}, \cdot)$ by assumption, and $f + \lambda g$ by combining this with the smoothness/Lipschitzness of f , as follows:

Lemma A.1. *If $\lambda \geq \max\{\frac{2L_1^g}{\mu_g}, \frac{L_0^f}{L_0^g}\}$ then for all $\mathbf{x} \in \mathcal{X}$: $f(\mathbf{x}, \cdot) + \lambda g(\mathbf{x}, \cdot)$ is $\frac{\lambda\mu_g}{2}$ strongly-convex, and moreover for all $i \in [n]$: $f_i(\mathbf{x}, \cdot) + \lambda g_i(\mathbf{x}, \cdot)$ is $2\lambda L_0^g$ -Lipschitz.*

We therefore invoke the following guarantee, which provides the optimal result for strongly-convex DP ERM via DP-Loc-GD (Algorithm 2).

Theorem A.2. *Suppose that $h : \mathbb{R}^{d_y} \rightarrow \mathbb{R}$ is a μ -strongly-convex function of the form $h(\mathbf{y}) = \frac{1}{n} \sum_{i=1}^n h(\mathbf{y}, \xi_i)$ where $h(\cdot, \xi_i)$ is L -Lipschitz for all $i \in [n]$. Suppose $\arg \min h =: \mathbf{y}^* \in \mathbb{B}(\mathbf{y}_0, R_0)$ and that $n \geq \frac{LR_0^{2/\log(d_y)}}{\mu\epsilon'}$. Then there is an assignment of parameters $M = \log_2 \log(\frac{\mu\epsilon'n}{L})$, $\sigma_{\text{GD}}^2 = \tilde{\mathcal{O}}(L^2/\epsilon'^2)$, $\eta_t = \frac{1}{\mu(t+1)}$, $T_{\text{GD}} = n^2$, $R_m = \tilde{\Theta}\left(\sqrt{\frac{R_{m-1}L}{\mu\epsilon'n}} + \frac{L\sqrt{d_y}}{\mu\epsilon'n}\right)$ such that running Algorithm 2 satisfies (ϵ', δ') -DP, and outputs \mathbf{y}_{out} such that $\|\mathbf{y}_{\text{out}} - \mathbf{y}^*\| = \tilde{\mathcal{O}}\left(\frac{L\sqrt{d_y}}{\mu n \epsilon'}\right)$ with probability at least $1 - \gamma$.*

Although the rate in Theorem A.2 appears in prior works such as (Bassily et al., 2014; Feldman et al., 2020), it is typically manifested through a bound in expectation (and in terms of function value) as opposed to with high probability, required for our purpose. We therefore, for the sake of completeness, provide a self-contained proof of Theorem A.2 in Appendix B.

Applied to the functions $g(\mathbf{x}_t, \cdot)$ and $f(\mathbf{x}_t, \cdot) + \lambda g(\mathbf{x}_t, \cdot)$, and invoking Lemma A.1, yields the following.

Corollary A.3. *If $\lambda \geq \max\{\frac{2L_1^g}{\mu_g}, \frac{L_0^f}{L_0^g}\}$, then $\tilde{\mathbf{y}}_t$ and $\tilde{\mathbf{y}}_t^\lambda$ (as appear in Line 3 of Algorithm 1) satisfy with probability at least $1 - \gamma$:*

$$\max\{\|\tilde{\mathbf{y}}_t - \mathbf{y}^*(\mathbf{x}_t)\|, \|\tilde{\mathbf{y}}_t^\lambda - \mathbf{y}^\lambda(\mathbf{x}_t)\|\} = \tilde{\mathcal{O}}\left(\frac{L_0^g \sqrt{d_y T}}{\epsilon \mu_g n}\right).$$

We are now ready to prove the main proposition of this section, which we restate below:

Lemma 3.4. *If $\lambda \geq \max\{\frac{2L_1^g}{\mu_g}, \frac{L_0^f}{L_0^g}, \frac{L_1^f}{L_1^g}\}$, then there is $\beta = \tilde{\mathcal{O}}\left(\frac{\lambda \kappa \sqrt{d_y T}}{\epsilon n}\right)$ such that with probability at least $1 - \gamma$, for all $t \in \{0, \dots, T - 1\}$:*

$$\|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - [\nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda (\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))]\| \leq \beta.$$

Proof of Lemma 3.4. As in Eq. (3), we note that by construction $\mathcal{L}_\lambda^*(\mathbf{x}) = \arg \min_{\mathbf{y}} \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y})$, therefore it holds that

$$\nabla \mathcal{L}_\lambda^*(\mathbf{x}) = \nabla_x f(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \lambda (\nabla_x g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \nabla_x g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))).$$

Denoting $\mathbf{g}_t = \nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda (\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))$, by the smoothness of f and g we see that

$$\|\mathbf{g}_t - \nabla \mathcal{L}_\lambda^*(\mathbf{x}_t)\| \leq L_1^f \|\tilde{\mathbf{y}}_t^\lambda - \mathbf{y}^\lambda(\mathbf{x}_t)\| + \lambda L_1^g \|\tilde{\mathbf{y}}_t^\lambda - \mathbf{y}^\lambda(\mathbf{x}_t)\| + \lambda L_1^g \|\tilde{\mathbf{y}}_t - \mathbf{y}^*(\mathbf{x}_t)\|.$$

Applying Corollary A.3 and union bounding over T , we can further bound the above as

$$\begin{aligned} \|\mathbf{g}_t - \nabla \mathcal{L}_\lambda^*(\mathbf{x}_t)\| &= \tilde{\mathcal{O}} \left(\frac{L_1^f L_0^g \sqrt{d_y T}}{\epsilon \mu_g n} + \frac{\lambda L_1^g L_0^g \sqrt{d_y T}}{\epsilon \mu_g n} + \frac{\lambda L_1^g L_0^g \sqrt{d_y T}}{\epsilon \mu_g n} \right) \\ &= \tilde{\mathcal{O}} \left(\frac{\lambda L_1^g L_0^g \sqrt{d_y T}}{\epsilon \mu_g n} \right) \\ &= \tilde{\mathcal{O}} \left(\frac{\lambda \ell \kappa \sqrt{d_y T}}{\epsilon n} \right), \end{aligned}$$

where the second bound holds for $\lambda \geq \frac{L_1^f}{L_1^g}$. \square

A.2 PROOF OF LEMMA 3.5

We start by providing two lemmas, both of which borrow ideas that appeared in the smoothness analysis of Chen et al. (2024), and are proved here for completeness.

Lemma A.4. $\mathbf{y}^\lambda(\mathbf{x}) : \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_y}$ is $\left(\frac{4L_1^g}{\mu_g}\right)$ -Lipschitz.

Proof of Lemma A.4. Differentiating $\nabla_y \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) = \mathbf{0}$ with respect the first argument gives

$$\nabla_{xy}^2 \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \nabla \mathbf{y}^\lambda(\mathbf{x}) \cdot \nabla_{yy} \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) = \mathbf{0},$$

hence

$$\nabla \mathbf{y}^\lambda(\mathbf{x}) = -\nabla_{xy}^2 \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) \cdot [\nabla_{yy} \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}))]^{-1}.$$

Noting that $\nabla_{xy}^2 \mathcal{L}_\lambda \preceq 2\lambda L_1^g$ and $\nabla_{yy}^2 \mathcal{L}_\lambda \succeq \lambda \mu_g / 2$ everywhere, hence $[\nabla_{yy}^2 \mathcal{L}_\lambda]^{-1} \preceq 2 / \lambda \mu_g$ we get that

$$\|\nabla \mathbf{y}^\lambda(\mathbf{x})\| \leq 2\lambda L_1^g \cdot \frac{2}{\lambda \mu_g} = \frac{4L_1^g}{\mu_g}.$$

\square

Lemma A.5. For all $\mathbf{x} \in \mathcal{X}$: $\|\mathbf{y}^\lambda(\mathbf{x}) - \mathbf{y}^*(\mathbf{x})\| \leq \frac{L_0^f}{\lambda \mu_g}$.

Proof of Lemma A.5. First, note that by definition of $\mathbf{y}^\lambda(\mathbf{x})$ it holds that

$$\mathbf{0} = \nabla_y \mathcal{L}_\lambda(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) = \nabla_y f(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \lambda \nabla_y g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})),$$

hence

$$\nabla_y g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) = -\frac{1}{\lambda} \cdot \nabla_y f(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})),$$

so in particular by the Lipschitz assumption on f we see that

$$\|\nabla_y g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}))\| \leq \frac{L_0^f}{\lambda}.$$

By invoking the μ -strong convexity of g we further get

$$\|\mathbf{y}^\lambda(\mathbf{x}) - \mathbf{y}^*(\mathbf{x})\| \leq \frac{1}{\mu} \left\| \nabla_y g(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \underbrace{\nabla_y g(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))}_{=0} \right\| \leq \frac{L_0^f}{\lambda \mu}.$$

\square

We are now ready to prove the main result of this section, restated below.

Lemma 3.5. $\mathcal{L}_{\lambda,i}^*$ is $\mathcal{O}(\ell \kappa)$ -Lipschitz.

864 *Proof of Lemma 3.5.* For all $\mathbf{x} \in \mathcal{X}$ it holds that

$$\begin{aligned} 865 \quad \|\nabla \mathcal{L}_{\lambda,i}^*(\mathbf{x})\| &= \|\nabla_x f_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) + \lambda [\nabla_x g_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \nabla_x g_i(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))]\| \\ 866 \quad &\leq \|\nabla_x f_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}))\| + \lambda \|\nabla_x g_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \nabla_x g_i(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))\|, \end{aligned} \quad (4)$$

868 thus we will bound each of the summands above.

869 For the first term, since \mathbf{y}^λ is $\frac{4L_1^g}{\mu_g}$ Lipschitz according to Lemma A.4, and f_i is L_0^f -Lipschitz by
870 assumption, the chain rule yields the bound

$$872 \quad \|\nabla_x f_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x}))\| \leq \frac{4L_1^g L_0^f}{\mu_g} \leq 4\ell\kappa. \quad (5)$$

874 As to the second term, since g_i is L_1^g -smooth, we use Lemma A.5 and get that

$$876 \quad \lambda \|\nabla_x g_i(\mathbf{x}, \mathbf{y}^\lambda(\mathbf{x})) - \nabla_x g_i(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))\| \leq \lambda L_1^g \|\mathbf{y}^\lambda(\mathbf{x}) - \mathbf{y}^*(\mathbf{x})\| \leq \frac{L_1^g L_0^f}{\mu_g} \leq \ell\kappa. \quad (6)$$

878 Plugging Eqs. (5) and (6) into Eq. (4) completes the proof.

879 \square

881 A.3 PROOF OF PROPOSITION 3.6

882 As $\nu_0, \dots, \nu_{T-1} \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2 I)$, a standard Gaussian norm bound (cf. Vershynin 2018, Theorem
883 3.1.1) ensures that with probability at least $1 - \gamma$, for all $t \in \{0, 1, \dots, T-1\}$: $\|\nu_t\|^2 \lesssim$
884 $d\sigma^2 \log(T/\gamma) \lesssim \frac{\alpha^2}{64}$. We therefore condition the rest of the proof on the highly probable event
885 under which this uniform norm bound indeed holds. We continue by introducing some notation. We
886 denote $\tilde{\nabla}_t = \tilde{\nabla}h(\mathbf{x}_t) + \nu_t$, and $\delta_t := \tilde{\nabla}_t - \nabla h(\mathbf{x}_t)$. We further denote

$$\begin{aligned} 889 \quad \mathbf{x}_t^+ &:= \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \langle \nabla h(\mathbf{x}_t), \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{x}_t - \mathbf{u}\|^2 \right\}, \\ 891 \quad \mathcal{G}_t &:= \frac{1}{\eta} (\mathbf{x}_t - \mathbf{x}_t^+), \\ 892 \quad \rho_t &:= \frac{1}{\eta} (\mathbf{x}_t - \mathbf{x}_{t+1}). \end{aligned}$$

895 Note that by construction,

$$897 \quad \mathcal{G}_t = \mathcal{G}_{h,\eta}(\mathbf{x}_t) := \frac{1}{\eta} (\mathbf{x}_t - \mathcal{P}_{\nabla h,\eta}(\mathbf{x}_t)), \quad \mathcal{P}_{\nabla h,\eta}(\mathbf{x}_t) := \arg \min_{\mathbf{u} \in \mathcal{X}} \left[\langle \nabla h(\mathbf{x}_t), \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}_t\|^2 \right],$$

898 and that $\mathcal{G}_{t,\text{out}}$ is precisely the quantity we aim to bound. We start by proving some useful lemmas
899 regarding the quantities defined above.

901 **Lemma A.6.** *Under the event that $\|\nu_t\|^2 \leq \frac{\alpha^2}{64}$ for all t , it holds that $\|\delta_t\| \leq \frac{\alpha}{4}$.*

902 *Proof.* By our assumptions on $\beta, \|\nu_t\|$, we get that

$$904 \quad \|\delta_t\| \leq \|\tilde{\nabla}h(\mathbf{x}_t) - \nabla h(\mathbf{x}_t)\| + \|\nu_t\| \leq \beta + \frac{\alpha}{8} \leq \frac{\alpha}{4}.$$

906 \square

907 **Lemma A.7.** *It holds that $\langle \tilde{\nabla}_t, \rho_t \rangle \geq \|\rho_t\|^2$.*

908 *Proof.* By definition, $\mathbf{x}_{t+1} = \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \langle \tilde{\nabla}_t, \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{x}_t - \mathbf{u}\|^2 \right\}$. Hence, by the first-order
909 optimality criterion, for any $\mathbf{u} \in \mathcal{X}$:

$$912 \quad \left\langle \tilde{\nabla}_t + \frac{1}{\eta} (\mathbf{x}_{t+1} - \mathbf{x}_t), \mathbf{u} - \mathbf{x}_{t+1} \right\rangle \geq 0.$$

914 In particular, setting $\mathbf{u} = \mathbf{x}_t$ yields

$$915 \quad 0 \leq \left\langle \tilde{\nabla}_t + \frac{1}{\eta} (\mathbf{x}_{t+1} - \mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t+1} \right\rangle = \langle \tilde{\nabla}_t - \rho_t, \eta \rho_t \rangle = \eta \left(\langle \tilde{\nabla}_t, \rho_t \rangle - \|\rho_t\|^2 \right),$$

916 which proves the claim since $\eta > 0$.

917 \square

With the lemmas above in hand, we are now ready to prove Proposition 3.6. Note that by construction, the algorithm returns the index t with minimal $\|\rho_t\|$. Further note that $\|\rho_t - \mathcal{G}_t\| \leq \|\delta_t\|$ by Lemma 2.4, thus

$$\|\mathcal{G}_t\| \leq \|\rho_t\| + \|\delta_t\| \leq \|\rho_t\| + \frac{\alpha}{4}, \quad (7)$$

where the last inequality is due to Lemma A.6, hence it suffices to bound $\|\rho_{t_{\text{out}}}\|$ (which is the quantity measured by the stopping criterion). To that end, by smoothness, we have for any $t \in \{0, 1, \dots, T-2\}$:

$$\begin{aligned} h(\mathbf{x}_{t+1}) &\leq h(\mathbf{x}_t) + \langle \nabla h(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{L}{2} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\ &= h(\mathbf{x}_t) - \eta \langle \nabla h(\mathbf{x}_t), \rho_t \rangle + \frac{L\eta^2}{2} \|\rho_t\|^2 \\ &= h(\mathbf{x}_t) - \eta \langle \tilde{\nabla}_t, \rho_t \rangle + \frac{L\eta^2}{2} \|\rho_t\|^2 + \eta \langle \delta_t, \rho_t \rangle \\ &\leq h(\mathbf{x}_t) - \eta \|\rho_t\|^2 + \frac{L\eta^2}{2} \|\rho_t\|^2 + \eta \|\delta_t\| \cdot \|\rho_t\|, \end{aligned}$$

where the last inequality followed by applying Lemma A.7 and Cauchy-Schwarz. Rearranging, and recalling that $\eta = \frac{1}{2L}$, hence $1 < 2 - L\eta$ and also $\frac{1}{\eta} = 2L$, we get that

$$\|\rho_t\|^2 - 2\|\delta_t\| \cdot \|\rho_t\| \leq (2 - L\eta) \|\rho_t\|^2 - 2\|\delta_t\| \cdot \|\rho_t\| \leq \frac{2(h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}))}{\eta} = 4L(h(\mathbf{x}_t) - h(\mathbf{x}_{t+1})).$$

Summing over $t \in \{0, 1, \dots, T-1\}$, using the telescoping property of the right hand side, and dividing by T gives that

$$\frac{1}{T} \sum_{t=0}^{T-1} \|\rho_t\| (\|\rho_t\| - 2\|\delta_t\|) \leq \frac{4L(h(\mathbf{x}_0) - \inf h)}{T}. \quad (8)$$

Note that if for some $t \in \{0, 1, \dots, T-1\}$: $\|\rho_t\| \leq \frac{3\alpha}{4}$ then we have proved our desired claim by Eq. (7) and the fact that $\|\rho_{t_{\text{out}}}\| = \min_t \|\rho_t\|$ by definition. On the other hand, assuming that $\|\rho_t\| > \frac{3\alpha}{4}$ for all t , invoking Lemma A.6, we see that $\|\rho_t\| - 2\|\delta_t\| \geq \|\rho_t\| - \frac{\alpha}{2} \geq \frac{1}{3}\|\rho_t\|$, which implies $\|\rho_t\| (\|\rho_t\| - 2\|\delta_t\|) \geq \frac{1}{3}\|\rho_t\|^2$. Combining this with Eq. (8) yields

$$\|\rho_{t_{\text{out}}}\|^2 = \min_{t \in \{0, 1, \dots, T-1\}} \|\rho_t\|^2 \leq \frac{1}{T} \sum_{t=0}^{T-1} \|\rho_t\|^2 \leq \frac{12L(h(\mathbf{x}_0) - \inf h)}{T},$$

and the right side is bounded by $\frac{9\alpha^2}{16}$ for $T = \mathcal{O}\left(\frac{L(h(\mathbf{x}_0) - \inf h)}{\alpha^2}\right)$, finishing the proof by Eq. (7).

A.4 PROOF OF THEOREM 3.1

We start by proving the privacy guarantee. Since $\mathcal{L}_{\lambda, i}^*$ is $\mathcal{O}(\ell\kappa)$ -Lipschitz by Lemma 3.5, the sensitivity of $\nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda (\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))$ is at most $\mathcal{O}(\ell\kappa)$. Hence, by setting $\sigma^2 = C \frac{\ell^2 \kappa^2 \log(T/\delta) T}{\epsilon^2 n^2}$ for a sufficiently large absolute constant $C > 0$, $\tilde{\mathbf{g}}_t$ is $(\frac{\epsilon}{\sqrt{18T}}, \frac{\delta}{3(T+1)})$ -DP. By basic composition, since each iteration also runs $(\frac{\epsilon}{\sqrt{18T}}, \frac{\delta}{3(T+1)})$ -DP-Loc-GD twice, we see that each iteration of the algorithm is $(3 \cdot \frac{\epsilon}{\sqrt{18T}}, 3 \cdot \frac{\delta}{3(T+1)}) = (\frac{\epsilon}{\sqrt{2T}}, \frac{\delta}{(T+1)})$ -DP. Noting that under our parameter assignment $\frac{\epsilon}{\sqrt{T}} \ll 1$, by advanced composition we get that throughout T iterations, the algorithm is overall (ϵ, δ) -DP as claimed.

We turn to analyze the utility of the algorithm. It holds that

$$\begin{aligned} \|\mathcal{G}_{F, \eta}(\mathbf{x}_{t_{\text{out}}})\| &\leq \|\mathcal{G}_{F, \eta}(\mathbf{x}_{t_{\text{out}}}) - \mathcal{G}_{\mathcal{L}_{\lambda, \eta}^*}(\mathbf{x})\| + \|\mathcal{G}_{\mathcal{L}_{\lambda, \eta}^*}(\mathbf{x}_{t_{\text{out}}})\| \\ &\leq \|\nabla F(\mathbf{x}_{t_{\text{out}}}) - \nabla \mathcal{L}_{\lambda}^*(\mathbf{x})\| + \|\mathcal{G}_{\mathcal{L}_{\lambda, \eta}^*}(\mathbf{x}_{t_{\text{out}}})\| \\ &\lesssim \frac{\ell\kappa^3}{\lambda} + \|\mathcal{G}_{\mathcal{L}_{\lambda, \eta}^*}(\mathbf{x}_{t_{\text{out}}})\| \\ &\leq \frac{\alpha}{2} + \|\mathcal{G}_{\mathcal{L}_{\lambda, \eta}^*}(\mathbf{x}_{t_{\text{out}}})\|, \end{aligned} \quad (9)$$

where the second inequality is due to Lemma 2.4, the third due to Lemma 3.3.b, and the last by our assignment of λ . It therefore remains to bound $\|\mathcal{G}_{\mathcal{L}_\lambda^*, \eta}(\mathbf{x}_{t_{\text{out}}})\|$.

To that end, applying Proposition 3.6 to the function $h = \mathcal{L}_\lambda^*$, under our assignment of T — which accounts for the smoothness and initial sub-optimality bounds ensured by Lemma 3.3 — we get that $\|\mathcal{G}_{\mathcal{L}_\lambda^*, \eta}(\mathbf{x}_{t_{\text{out}}})\| \leq \frac{\alpha}{2}$, for α as small as

$$\alpha = \Theta\left(\max\{\beta, \sigma\sqrt{d_x \log(T/\gamma)}\}\right) \quad (10)$$

By Lemma 3.4, it holds that

$$\beta = \tilde{\mathcal{O}}\left(\frac{\lambda\ell\kappa\sqrt{d_y T}}{\epsilon n}\right) = \tilde{\mathcal{O}}\left(\frac{\ell^{3/2}\kappa^{11/2}\Delta_F^{1/2}\sqrt{d_y}}{\alpha^2\epsilon n}\right) \quad (11)$$

and we also have

$$\sigma\sqrt{d_x \log(T/\gamma)} = \tilde{\mathcal{O}}\left(\frac{\ell^{3/2}\kappa^{5/2}\Delta_F^{1/2}\sqrt{d_x}}{\alpha\epsilon n}\right). \quad (12)$$

Plugging (11) and (12) back into Eq. (10), and solving for α , completes the proof.

A.5 PROOF OF THEOREM 4.1

Throughout this section, we abbreviate $b = b_{\text{out}}$. We will need the following lemma, which is the mini-batch analogue of Lemma 3.4 from the full-batch setting.

Lemma A.8. *If $\lambda \geq \max\{\frac{2L_1^g}{\mu_g}, \frac{L_0^f}{L_0^g}, \frac{L_1^f}{L_1^g}\}$, then there is $\beta_b = \tilde{\mathcal{O}}\left(\frac{\lambda\ell\kappa\sqrt{d_y T}}{\epsilon n} + \frac{\ell\kappa}{b}\right)$ such that with probability at least $1 - \gamma/2$, $\mathbf{g}_t^B := \nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; B_t) + \lambda(\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; B_t) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))$ satisfies for all $t \in \{0, \dots, T-1\}$: $\|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - \mathbf{g}_t^B\| \leq \beta_b$.*

Proof of Lemma A.8. It holds that

$$\|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - \mathbf{g}_t^B\| \leq \|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - \mathbb{E}[\mathbf{g}_t^B]\| + \|\mathbf{g}_t^B - \mathbb{E}[\mathbf{g}_t^B]\|.$$

To bound the first summand, note that $\mathbb{E}[\mathbf{g}_t^B] = \nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) + \lambda(\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t))$, and therefore with probability at least $1 - \gamma/4$:

$$\|\nabla \mathcal{L}_\lambda^*(\mathbf{x}_t) - \mathbb{E}[\mathbf{g}_t^B]\| = \mathcal{O}\left(\frac{\lambda L_1^g L_0^g \sqrt{d_y T}}{\epsilon \mu_g n}\right) = \mathcal{O}\left(\frac{\lambda \ell \kappa \sqrt{d_y T}}{\epsilon n}\right),$$

following the same proof as Lemma 3.4 in Appendix A.1, by replacing Theorem A.2 by the mini-batch Theorem B.1 (whose guarantee holds regardless of the inner batch size).

To bound the second summand, note that $\|\nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; \xi) + \lambda(\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; \xi) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t; \xi))\| \leq M = \mathcal{O}(\ell\kappa)$ for every $\xi \in \Xi$, by Lemma 3.5. Hence, \mathbf{g}_t^B is the average of b independent vectors bounded by M , all with the same mean, and therefore a standard concentration bound (cf. Jin et al. 2019) ensures that $\|\mathbf{g}_t^B - \mathbb{E}[\mathbf{g}_t^B]\| = \tilde{\mathcal{O}}(M/b)$ with probability at least $1 - \gamma/4$, which completes the proof. \square

We can now prove the main mini-batch result:

Proof of Theorem 4.1. We start by proving the privacy guarantee. Since $\mathcal{L}_{\lambda, i}^*$ is $\mathcal{O}(\ell\kappa)$ -Lipschitz by Lemma 3.5, the sensitivity of $\nabla_x f(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; B_t) + \lambda(\nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t^\lambda; B_t) - \nabla_x g(\mathbf{x}_t, \tilde{\mathbf{y}}_t; B_t))$ is at most $\mathcal{O}(\ell\kappa)$. Accordingly, the “unamplified” Gaussian mechanism (Lemma 2.2) ensures $(\tilde{\epsilon}, \tilde{\delta})$ -DP for $\tilde{\epsilon} = \tilde{\Theta}\left(\frac{\ell\kappa}{b\sigma}\right)$, and hence is amplified (Lemma 2.3) to (ϵ_0, δ_0) -DP for $\epsilon_0 = \tilde{\Theta}\left(\frac{\ell\kappa}{b\sigma} \cdot \frac{b}{n}\right) = \frac{\epsilon}{\sqrt{18T}}$, the last holding for sufficiently large $\sigma^2 = \tilde{\Theta}\left(\frac{\ell^2\kappa^2 T}{\epsilon^2 n^2}\right)$, and for $\delta_0 = \frac{\delta}{3(T+1)}$. Therefore, basic composition shows that each iteration of the algorithm is $(3 \cdot \frac{\epsilon}{\sqrt{18T}}, 3 \frac{\delta}{3(T+1)}) = (\frac{\epsilon}{\sqrt{2T}}, \frac{\delta}{T+1})$ -DP.

Since $\epsilon/\sqrt{T} \ll 1$ under our parameter assignment, advanced composition over the T iterations yields the (ϵ, δ) -DP guarantee.

We turn to analyze the utility of the algorithm. It holds that

$$\begin{aligned} \|\mathcal{G}_{F,\eta}(\mathbf{x}_{t_{\text{out}}})\| &\leq \|\mathcal{G}_{F,\eta}(\mathbf{x}_{t_{\text{out}}}) - \mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x})\| + \|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\| \\ &\leq \|\nabla F(\mathbf{x}_{t_{\text{out}}}) - \nabla \mathcal{L}_\lambda^*(\mathbf{x})\| + \|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\| \\ &\lesssim \frac{\ell\kappa^3}{\lambda} + \|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\| \\ &\leq \frac{\alpha}{2} + \|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\|, \end{aligned} \quad (13)$$

where the second inequality is due to Lemma 2.4, the third due to Lemma 3.3, and the last by our assignment of λ . It therefore remains to bound $\|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\|$.

To that end, applying Proposition 3.6 to the function $h = \mathcal{L}_\lambda^*$, under our assignment of T — which accounts for the smoothness and initial sub-optimality bounds ensured by Lemma 3.3 — we get that $\|\mathcal{G}_{\mathcal{L}_\lambda^*,\eta}(\mathbf{x}_{t_{\text{out}}})\| \leq \frac{\alpha}{2}$, for α as small as

$$\alpha = \Theta\left(\max\{\beta_b, \sigma\sqrt{d_x \log(T/\gamma)}\}\right) \quad (14)$$

By Lemma A.8, it holds that

$$\beta_b = \tilde{\mathcal{O}}\left(\frac{\lambda\ell\kappa\sqrt{d_y T}}{\epsilon n} + \frac{\ell\kappa}{b}\right) = \tilde{\mathcal{O}}\left(\frac{\ell^{3/2}\kappa^{11/2}\Delta_F^{1/2}\sqrt{d_y}}{\alpha^2\epsilon n} + \frac{\ell\kappa}{b}\right), \quad (15)$$

and we also have

$$\sigma\sqrt{d_x \log(T/\gamma)} = \tilde{\mathcal{O}}\left(\frac{\ell^{3/2}\kappa^{5/2}\Delta_F^{1/2}\sqrt{d_x}}{\alpha\epsilon n}\right) \quad (16)$$

Plugging (15) and (16) back into Eq. (14), and solving for α , completes the proof. \square

A.6 PROOF OF THEOREM 5.1

We first need a simple lemma that immediately follows from our assumptions, and Eq. (1):

Lemma A.9. *Under Assumptions 2.5 and 2.6, $F(\cdot; \xi)$ is G -Lipschitz, for $G = \mathcal{O}(\ell\kappa)$.*

Accordingly, the main tool that will allow us to obtain generalization guarantees, is the following uniform convergence result in terms of gradients:

Lemma A.10 (Mei et al., 2018, Theorem 1). *Suppose $\bar{\mathcal{X}} \subset \mathbb{R}_x^d$ is a subset of bounded diameter $\text{diam}(\bar{\mathcal{X}}) \leq D$, and that $S \sim \mathcal{P}^n$. Then with probability at least $1 - \gamma$ for all $\mathbf{x} \in \bar{\mathcal{X}}$: $\|\nabla F_{\mathcal{P}}(\mathbf{x}) - \nabla F_S(\mathbf{x})\| = \tilde{\mathcal{O}}\left(G\sqrt{d_x \log(D/\gamma)/n}\right)$, where G is the Lipschitz constant of F .⁵*

Proof of Theorem 5.1. With probability at least $1 - \gamma/2$ it holds that

$$\begin{aligned} \|\mathcal{G}_{F_{\mathcal{P}},\eta}(\mathbf{x}_{\text{out}})\| &\leq \|\mathcal{G}_{F_S,\eta}(\mathbf{x}_{\text{out}})\| + \|\mathcal{G}_{F_{\mathcal{P}},\eta}(\mathbf{x}_{\text{out}}) - \mathcal{G}_{F_S,\eta}(\mathbf{x}_{\text{out}})\| \\ &\leq \|\mathcal{G}_{F_S,\eta}(\mathbf{x}_{\text{out}})\| + \|\nabla F_{\mathcal{P}}(\mathbf{x}_{\text{out}}) - \nabla F_S(\mathbf{x}_{\text{out}})\| \\ &= \|\mathcal{G}_{F_S,\eta}(\mathbf{x}_{\text{out}})\| + \tilde{\mathcal{O}}\left(\ell\kappa\sqrt{\frac{d_x}{n}}\right), \end{aligned}$$

where the second inequality is due to Lemma 2.4, and the last is by Lemma A.10 with the Lipschitz bound of Lemma A.9 and the domain bound $\|\mathbf{x}_{\text{out}} - \mathbf{x}_0\| \leq D$ for some sufficiently large D which is polynomial in all problem parameters (therefore only affecting log terms). The results then follow from Theorems 3.1 and 4.1. \square

⁵Mei et al. (2018) originally stated this for functions whose gradients are sub-Gaussian vectors. By Lemma A.9, the gradients are G -bounded, hence $\mathcal{O}(G)$ -sub-Gaussian.

B OPTIMAL DP ALGORITHM FOR STRONGLY-CONVEX OBJECTIVES

The goal of this appendix is to provide a self contained analysis of a DP algorithm for strongly-convex optimization which achieves the optimal convergence rate with a high probability guarantee. Any such algorithm can be used as the inner loop in our DP bilevel algorithm.

In particular, we analyze *localized* DP (S)GD. Although it would have been more natural to apply DP-SGD, this seems (at least according to our analysis) to yield an inferior rate with respect to the required high probability guarantee.⁶ Indeed, for DP-(S)GD, previous works (such as Bassily et al. 2014) typically provide bounds in expectation, and then convert them into high-probability bounds via a black-box reduction, which applies several runs and selects the best run via the private noisy-min (i.e. Laplace mechanism). The additional error incurred by this selection is of order $\frac{1}{n}$, which translates to $\frac{1}{\sqrt{n}}$ in terms of distance to the optimum, thus spoiling the fast rate of $\frac{1}{n}$ otherwise achieved in expectation for strongly-convex objectives. We therefore resort to localization (Feldman et al., 2020): by running projected-(S)GD over balls with shrinking radii, applying martingale concentration bounds enables us to show that the distance to optimum shrinks as $R_{m+1} \lesssim \sqrt{\frac{R_m}{n}} + \frac{1}{n}$, and thus with negligible overhead we eventually recover the optimal fast rate $R_M \lesssim \frac{1}{n}$ with high probability. Our analysis differs than previous localization analyses, as it does not require adapting the noise-level and step sizes throughout the rounds.

We prove the following (which easily implies also the full-batch Theorem A.2):

Theorem B.1. *Suppose that $h : \mathbb{R}^{d_y} \times \Xi \rightarrow \mathbb{R}$ is a μ -strongly-convex function of the form $h(\mathbf{y}) = \frac{1}{n} \sum_{i=1}^n h(\mathbf{y}, \xi_i)$ where $h(\cdot, \xi_i)$ is L -Lipschitz for all $i \in [n]$. Suppose $\arg \min h =: \mathbf{y}^* \in \mathbb{B}(\mathbf{y}_0, R_0)$, and that $n \geq \frac{LR_0^{\frac{2}{\log(d_y)}}}{\mu\epsilon'}$. Then given any batch size $b \in \{1, \dots, n\}$, there is an assignment of parameters $M = \log_2 \log(\frac{\mu\epsilon'n}{L})$, $\sigma_{\text{SGD}}^2 = \tilde{\mathcal{O}}\left(\frac{L^2}{\epsilon'^2}\right)$, $\eta_t = \frac{1}{\mu(t+1)}$, $T_{\text{SGD}} = n^2$, $R_m = \tilde{\mathcal{O}}\left(\sqrt{\frac{R_{m-1}L}{\mu\epsilon'n}} + \frac{L\sqrt{d_y}}{\mu\epsilon'n}\right)$ such that running Algorithm 4 satisfies (ϵ, δ) -DP, and outputs \mathbf{y}_{out} such that $\|\mathbf{y}_{\text{out}} - \mathbf{y}^*\| = \tilde{\mathcal{O}}\left(\frac{L\sqrt{d_y}}{\mu n \epsilon}\right)$ with probability at least $1 - \gamma$.*

Proof of Theorem B.1. We start by proving the privacy guarantee. By the Lipschitz assumption, the sensitivity of $\nabla h(\cdot; B_t)$ is at most $\frac{2L}{b}$, thus the “unamplified” Gaussian mechanism (Lemma 2.2) ensures $(\tilde{\epsilon}, \tilde{\delta})$ -DP with $\tilde{\epsilon} = \tilde{\mathcal{O}}\left(\frac{L}{b\sigma}\right) = \tilde{\mathcal{O}}\left(\frac{\epsilon'}{b}\right)$, and hence is amplified (Lemma 2.3) to (ϵ_0, δ_0) -DP for $\epsilon_0 = \tilde{\mathcal{O}}\left(\frac{\epsilon'}{b} \cdot \frac{b}{n}\right) = \tilde{\mathcal{O}}\left(\frac{\epsilon'}{n}\right) = \tilde{\mathcal{O}}\left(\frac{\epsilon'}{\sqrt{T}}\right)$. Advanced composition (Lemma 2.1) therefore ensures that the overall algorithm is (ϵ', δ') -DP (note that this uses the fact that $M = \tilde{\mathcal{O}}(1)$).

We turn to prove the utility of the algorithm. We first show that for all m :

$$\Pr[\mathbf{y}^* \in \mathbb{B}(\mathbf{y}_0^m, R_m)] \geq 1 - \frac{m\gamma}{M}. \quad (17)$$

We prove this by induction over m . The base case $m = 0$ follows by the assumption $\mathbf{y}^* \in \mathbb{B}(\mathbf{y}_0, R_0)$. Denoting $\mathbf{e}_t^m := \nabla h(\mathbf{y}_t^m; B_t) - \nabla h(\mathbf{y}_t^m)$, using the inductive hypothesis that $\mathbf{y}^* \in \mathbb{B}(\mathbf{y}_0^m, R_m)$ with probability at least $1 - \frac{m\gamma}{M}$, under this probably event we get

⁶Even if we would have sought only expectation bounds with respect to the hyperobjective, the high probability bound with respect to the inner problem is key to being able to argue about the gradient inexactness thereafter.

$$\begin{aligned}
\|\mathbf{y}_{t+1}^m - \mathbf{y}^*\|^2 &= \left\| \text{Proj}_{\mathbb{B}(\mathbf{y}_0^m, R_m)} [\mathbf{y}_t^m - \eta_t (\nabla h(\mathbf{y}_t^m; B_t^m) + \nu_t^m)] - \mathbf{y}^* \right\|^2 \\
&\leq \|\mathbf{y}_t^m - \eta_t (\nabla h(\mathbf{y}_t^m; B_t^m) + \nu_t^m) - \mathbf{y}^*\|^2 \\
&= \|\mathbf{y}_t^m - \mathbf{y}^*\|^2 - 2\eta_t \langle \mathbf{y}_t^m - \mathbf{y}^*, \nabla h(\mathbf{y}_t^m; B_t^m) + \nu_t^m \rangle + \eta_t^2 \|\nabla h(\mathbf{y}_t^m; B_t^m) + \nu_t^m\|^2 \\
&\leq \|\mathbf{y}_t^m - \mathbf{y}^*\|^2 - 2\eta_t \langle \mathbf{y}_t^m - \mathbf{y}^*, \nabla h(\mathbf{y}_t) + \mathbf{e}_t^m + \nu_t^m \rangle + 2\eta_t^2 \left(\|\nu_t^m\|^2 + \|\nabla h(\mathbf{y}_t)\|^2 \right) \\
&= \|\mathbf{y}_t^m - \mathbf{y}^*\|^2 - 2\eta_t \langle \mathbf{y}_t^m - \mathbf{y}^*, \nabla h(\mathbf{y}_t^m) \rangle \\
&\quad - 2\eta_t \langle \mathbf{y}_t^m - \mathbf{y}^*, \mathbf{e}_t^m + \nu_t^m \rangle + 2\eta_t^2 \left(\|\nu_t^m\|^2 + \|\nabla h(\mathbf{y}_t)\|^2 \right).
\end{aligned}$$

Rearranging, and using the strong convexity and Lipschitz assumptions, we see that

$$\begin{aligned}
h(\mathbf{y}_t^m) - h(\mathbf{y}^*) &\leq \langle \mathbf{y}_t^m - \mathbf{y}^*, \nabla h(\mathbf{y}_t^m) \rangle - \frac{\mu}{2} \|\mathbf{y}_t^m - \mathbf{y}^*\|^2 \\
&\leq \left(\frac{1}{2\eta_t} - \frac{\mu}{2} \right) \|\mathbf{y}_t^m - \mathbf{y}^*\|^2 - \frac{1}{2\eta_t} \|\mathbf{y}_{t+1}^m - \mathbf{y}^*\|^2 \\
&\quad - \langle \mathbf{y}_t^m - \mathbf{y}^*, \mathbf{e}_t^m + \nu_t^m \rangle + \eta_t \left(\|\nu_t^m\|^2 + L^2 \right).
\end{aligned}$$

Averaging over t and using $\eta_t = \frac{1}{\mu(t+1)}$, which satisfies $\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \mu \right) \leq 0$ and $\frac{1}{T} \sum_{t=0}^{T-1} \eta_t \lesssim \frac{\log T}{\mu T}$, by Jensen's inequality, overall we get with probability at least $1 - \frac{m\gamma}{M}$:

$$\begin{aligned}
h(\mathbf{y}_0^{m+1}) - h(\mathbf{y}^*) &= h \left(\frac{1}{T} \sum_{t=0}^{T-1} \mathbf{y}_t^m \right) - h(\mathbf{y}^*) \\
&\lesssim \underbrace{\left| \frac{1}{T} \sum_{t=0}^{T-1} \langle \mathbf{y}_t^m - \mathbf{y}^*, \mathbf{e}_t^m + \nu_t^m \rangle \right|}_{(I)} + \frac{L^2 \log T}{\mu T} + \frac{\log T}{\mu T} \underbrace{\sum_{t=0}^{T-1} \|\nu_t^m\|^2}_{(II)}. \quad (18)
\end{aligned}$$

We now apply concentration inequalities to bound (I) and (II) with high probability, for which we will use basic properties of sub-Gaussian distributions (cf. [Vershynin 2018](#), §3.4). To bound (I), note that for all t : $\mathbb{E} \mathbf{e}_t^m = \mathbb{E} \nu_t^m = \mathbf{0}$ and therefore $\mathbb{E} \langle \mathbf{y}_t^m - \mathbf{y}^*, \mathbf{e}_t^m + \nu_t^m \rangle = 0$. Moreover, $\mathbf{e}_t^m = \frac{1}{b} \sum_{\xi \in B_t^m} (\nabla h(\mathbf{x}_t^m; \xi) - \nabla h(\mathbf{y}_t^m))$ is the average of b independent vectors with norm bounded by at most $2L$, while $\nu_t^m \sim \mathcal{N}(\mathbf{0}, \sigma_{\text{SGD}}^2 I_{d_y}) = \mathcal{N}(\mathbf{0}, \tilde{\mathcal{O}}(\frac{L^2}{\epsilon'^2}) \cdot I_{d_y})$, and also $\|\mathbf{y}_t^m - \mathbf{y}^*\| \leq R_m$ by the inductive hypothesis. By combining all of these observations, we see that $\langle \mathbf{y}_t^m - \mathbf{y}^*, \mathbf{e}_t^m + \nu_t^m \rangle$ is a $\mathcal{O}(R_m \cdot (\frac{L}{b} + \frac{L}{\epsilon'})) = \mathcal{O}(\frac{R_m L}{\epsilon'})$ -sub-Gaussian random variable. By Azuma's inequality for sub-Gaussians ([Shamir, 2011](#)), we get that with probability at least $1 - \frac{\gamma}{2M}$:

$$(I) = \tilde{\mathcal{O}} \left(\frac{R_m L \log(\gamma/M)}{\sqrt{T}} \right) = \tilde{\mathcal{O}} \left(\frac{R_m L}{\epsilon' n} \right). \quad (19)$$

To bound (II), by concentration of the Gaussian norm (cf. [Vershynin 2018](#), Theorem 3.1.1) and the union bound we can get that with probability at least $1 - \frac{\gamma}{2M}$:

$$(II) = \tilde{\mathcal{O}}(d_y \sigma_{\text{SGD}}^2) = \tilde{\mathcal{O}} \left(\frac{d_y L^2}{\epsilon'^2} \right). \quad (20)$$

Plugging Eqs. (19) and (20) into Eq. (18), we overall get that with probability at least $1 - \frac{m\gamma}{M} - 2 \cdot \frac{\gamma}{2M} = 1 - \frac{(m+1)\gamma}{M}$:

$$h(\mathbf{y}_0^{m+1}) - h(\mathbf{y}^*) = \tilde{\mathcal{O}} \left(\frac{R_m L}{\epsilon' n} + \frac{d_y L^2}{\mu \epsilon'^2 n^2} \right).$$

Applying the μ -strong-convexity of h , and sub-additivity of the square root, we get that

$$\|\mathbf{y}_0^{m+1} - \mathbf{y}^*\| \leq \sqrt{\frac{2(h(\mathbf{y}_0^{m+1}) - h(\mathbf{y}^*))}{\mu}} = \tilde{\mathcal{O}} \left(\sqrt{\frac{R_m L}{\mu \epsilon' n}} + \frac{L \sqrt{d_y}}{\mu \epsilon' n} \right) \leq R_{m+1}.$$

We have therefore proven Eq. (17). In particular for $m = M$ we get that with probability at least $1 - \gamma$:

$$\|\mathbf{y}_{\text{out}} - \mathbf{y}^*\| \leq R_M, \quad (21)$$

hence it remains to bound R_M . We will prove, once again by induction over m , that

$$R_m = \tilde{\mathcal{O}} \left(R_0^{\frac{1}{2^m}} \left(\frac{L}{\mu\epsilon'n} \right)^{1 - \frac{1}{2^m}} + \frac{L}{\mu\epsilon'n} \sum_{i=1}^m d_y^{\frac{1}{2^i}} \right). \quad (22)$$

The base $m = 0$ simply follows since the left hand side in Eq. (22) reduces to R_0 . Denoting $A := \frac{L}{\mu\epsilon'n}$, by our assignment of R_{m+1} , the induction hypothesis and sub-additivity of the square root we get:

$$\begin{aligned} R_{m+1} &= \tilde{\mathcal{O}} \left(\sqrt{R_m A} + A \sqrt{d_y} \right) \\ &= \tilde{\mathcal{O}} \left(A^{1/2} \left(R_0^{\frac{1}{2^{m+1}}} A^{\frac{1}{2} - \frac{1}{2^{m+1}}} + A^{1/2} \sum_{i=1}^m d_y^{\frac{1}{2^{i+1}}} \right) + A d_y^{1/2} \right) \\ &= \tilde{\mathcal{O}} \left(R_0^{\frac{1}{2^{m+1}}} A^{1 - \frac{1}{2^{m+1}}} + A \sum_{i=1}^{m+1} d_y^{\frac{1}{2^i}} \right), \end{aligned}$$

therefore proving Eq. (22). In particular, for $m = M = \log_2 \log(\frac{\mu\epsilon'n}{L})$, which satisfies $\frac{1}{2^M} = \frac{1}{\log(\frac{\mu\epsilon'n}{L})} = \frac{1}{\log(1/A)}$ we get

$$\begin{aligned} R_M &= \tilde{\mathcal{O}} \left(R_0^{\frac{1}{\log(1/A)}} A^{1 + \frac{1}{\log(A)}} + M A d_y^{1/2} \right) \\ &= \tilde{\mathcal{O}} \left(R_0^{\frac{1}{\log(1/A)}} A + A d_y^{1/2} \right) \\ &= \tilde{\mathcal{O}} \left(R_0^{\frac{1}{\log(\frac{\mu\epsilon'n}{L})}} \frac{L}{\mu\epsilon'n} + \frac{L d_y^{1/2}}{\mu\epsilon'n} \right) \\ &= \tilde{\mathcal{O}} \left(\frac{L \sqrt{d_y}}{\mu\epsilon'n} \right), \end{aligned}$$

where the last follows from our assumption on n . This completes the proof by Eq. (21). \square

C AUXILIARY LEMMA

Lemma 2.4. For any $\mathbf{x}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^d$, $\eta > 0$: $\|\mathcal{G}_{\mathbf{v},\eta}(\mathbf{x}) - \mathcal{G}_{\mathbf{w},\eta}(\mathbf{x})\| \leq \|\mathbf{v} - \mathbf{w}\|$.

Proof of Lemma 2.4. The proof is due to Ghadimi et al. (2016). By definition,

$$\begin{aligned} \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) &= \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \langle \mathbf{v}, \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{x} - \mathbf{u}\|^2 \right\}, \\ \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) &= \arg \min_{\mathbf{u} \in \mathcal{X}} \left\{ \langle \mathbf{w}, \mathbf{u} \rangle + \frac{1}{2\eta} \|\mathbf{x} - \mathbf{u}\|^2 \right\}, \end{aligned}$$

hence by first order optimality criteria, for any $\mathbf{u} \in \mathcal{X}$:

$$\begin{aligned} \left\langle \mathbf{v} + \frac{1}{\eta} (\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathbf{x}), \mathbf{u} - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) \right\rangle &\geq 0, \\ \left\langle \mathbf{w} + \frac{1}{\eta} (\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathbf{x}), \mathbf{u} - \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) \right\rangle &\geq 0. \end{aligned}$$

Plugging $\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x})$ into the first inequality above, and $\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x})$ into the second, shows that

$$\begin{aligned} 0 &\leq \left\langle \mathbf{v} + \frac{1}{\eta} (\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathbf{x}), \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) \right\rangle, \\ 0 &\leq \left\langle \mathbf{w} + \frac{1}{\eta} (\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathbf{x}), \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) \right\rangle = \left\langle -\mathbf{w} + \frac{1}{\eta} (\mathbf{x} - \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x})), \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) \right\rangle. \end{aligned}$$

1242 Summing the two inequalities yields
 1243

$$\begin{aligned}
 1244 \quad 0 &\leq \left\langle \mathbf{v} - \mathbf{w} + \frac{1}{\eta}(\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x})), \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) \right\rangle \\
 1245 &= \langle \mathbf{v} - \mathbf{u}, \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) \rangle - \frac{1}{\eta} \|\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x})\|^2 \\
 1246 &\leq \|\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x})\| \left(\|\mathbf{v} - \mathbf{u}\| - \frac{1}{\eta} \|\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{v},\eta}(\mathbf{x})\| \right).
 \end{aligned}$$

1247 Hence,

$$\begin{aligned}
 1251 \quad \|\mathbf{v} - \mathbf{u}\| &\geq \frac{1}{\eta} \|\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathcal{P}_{\mathbf{w},\eta}(\mathbf{x})\| \\
 1252 &= \left\| \frac{1}{\eta} (\mathcal{P}_{\mathbf{v},\eta}(\mathbf{x}) - \mathbf{x}) - \frac{1}{\eta} (\mathcal{P}_{\mathbf{w},\eta}(\mathbf{x}) - \mathbf{x}) \right\| \\
 1253 &= \|\mathcal{G}_{\mathbf{v},\eta}(\mathbf{x}) - \mathcal{G}_{\mathbf{w},\eta}(\mathbf{x})\|.
 \end{aligned}$$

1254 □

1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295