Alternating Projected SGD for Equality-constrained Bilevel Optimization

Quan Xiao[⋆] **Han Shen**[⋆] **Wotao Yin**[†] **Tianyi Chen**[⋆]

[⋆]Rensselaer Polytechnic Institute

[†]Alibaba US, DAMO Academy

Abstract

Bilevel optimization, which captures the inherent nested structure of machine learning problems, is gaining popularity in many recent applications. Existing works on bilevel optimization mostly consider either the unconstrained problems or the constrained upper-level problems. In this context, this paper considers the stochastic bilevel optimization problems with equality constraints in both upper and lower levels. By leveraging the special structure of the equality constraints problem, the paper first presents an alternating projected SGD approach to tackle this problem and establishes the $\tilde{\mathcal{O}}(\epsilon^{-2})$ sample and iteration complexity that matches the state-of-the-art complexity of ALSET (Chen et al., 2021) for stochastic unconstrained bilevel problems. To further save the cost of projection, the paper presents an alternating projected SGD approach with lazy projection and establishes the $\tilde{\mathcal{O}}(\epsilon^{-2}/T)$ upper-level and $\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$ lower-level projection complexity of this new algorithm, where T is the upper-level projection interval. Application to federated bilevel optimization has been presented to showcase the performance of our algorithms. Our results demonstrate that equality-constrained bilevel optimization with strongly-convex lower-level problems can be solved as efficiently as stochastic single-level optimization problems. The code is available at `https://github.com/hanshen95/AiPOD`.

1 Introduction

Projected stochastic gradient descent (SGD) is a fundamental approach to solving large-scale constrained single-level machine learning problems. Specifically, to minimize $\mathbb{E}_\xi\left[\mathcal{L}(x;\xi)\right]$ over a given convex set $\mathcal{X}$, it generates the sequence $x^{k+1} = \mathrm{Proj}_{\mathcal{X}}(x^k - \alpha\nabla\mathcal{L}(x^k;\xi^k))$, where $\alpha > 0$ is the stepsize and $\nabla\mathcal{L}(x^k;\xi^k)$ is a stochastic gradient estimate of $\mathbb{E}_\xi\left[\mathcal{L}(x^k;\xi)\right]$. If $\mathbb{E}_\xi\left[\mathcal{L}(x;\xi)\right]$ is nonconvex, projected SGD requires a sample complexity of $\mathcal{O}(\epsilon^{-2})$ with $\mathcal{O}(1/\epsilon)$ batch size (Ghadimi et al., 2016). The requirement of $\mathcal{O}(1/\epsilon)$ batch size has been later relaxed in (Davis and Drusvyatskiy, 2019) using the Moreau envelope technique, and its convergence rate matches that of vanilla SGD.

However, recent machine learning applications often go beyond the single-level structure, including hyperparameter optimization (Maclaurin et al., 2015; Franceschi et al., 2017), meta learning (Finn et al., 2017), reinforcement learning (Sutton and Barto, 2018) and neural architecture search (Liu et al., 2019). While the nonasymptotic analysis of the alternating SGD for unconstrained bilevel optimization with strongly convex and smooth lower-level problems were well-understood (Ghadimi and Wang, 2018; Hong et al., 2020; Ji et al., 2021; Chen et al., 2021; Li et al., 2022a), to our best of knowledge, finite-time guarantee of alternating projected SGD on bilevel problems with both upper-level (UL) and lower-level (LL) constraints have not been investigated yet. In this context, a natural but important question is

Can we establish the $\tilde{\mathcal{O}}(\epsilon^{-2})$ iteration and sample complexity of alternating projected SGD for a family of bilevel problems with both UL and LL constraints?

We give an affirmative answer to this question for the following *stochastic bilevel optimization problems* with both UL and LL constraints, given by

$$\min_{x\in\mathcal{X}} \quad F(x) \triangleq \mathbb{E}_\xi[f(x, y^*(x);\xi)] \qquad \text{(upper)} \quad \text{(1a)}$$

$$\text{s.t.} \quad y^*(x) \triangleq \operatorname*{arg\,min}_{y\in\mathcal{Y}(x)} \mathbb{E}_\phi[g(x,y;\phi)] \qquad \text{(lower)} \quad \text{(1b)}$$

where ξ and ϕ are random variables, $\mathcal{X} = \{x \mid Bx = e\} \subset \mathbb{R}^{d_x}$ and $\mathcal{Y}(x) = \{y \mid Ay + h(x) = c\} \subset \mathbb{R}^{d_y}$ are closed convex set; $A \in \mathbb{R}^{m_y\times d_y}, B \in \mathbb{R}^{m_x\times d_x}, c \in \mathbb{R}^{m_y}, e \in \mathbb{R}^{m_x}, h : \mathbb{R}^{d_x} \to \mathbb{R}^{m_y}$; A and B are not necessarily full row or column rank and h can be nonlinear. In (1), the UL optimization problem depends on the solution of the LL optimization over y, and both the LL function and constraint set depend on the UL variable x. The equality constrained

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

bilevel problem (1) covers a wider class of applications than the unconstrained bilevel optimization, such as distributed bilevel optimization (Tarzanagh et al., 2022; Yang et al., 2022), hyperparameter optimization for optimal transport (Luise et al., 2018; Gould et al., 2022), and design of transportation networks (Marcotte, 1986; Alizadeh et al., 2013). When $A = 0, B = 0, h = 0, c = 0, e = 0$, the problem (1) reduces to the unconstrained stochastic bilevel problem (Ghadimi and Wang, 2018; Hong et al., 2020; Ji et al., 2021; Khanduri et al., 2021; Chen et al., 2021, 2022a).

Generically speaking, to solve (1), we can resort to alternating projected SGD method that performs

$$y^{k+1} = \mathrm{Proj}_{\mathcal{Y}(x^k)}\left(y^k - \beta h_g^k\right) \tag{2a}$$

$$x^{k+1} = \mathrm{Proj}_{\mathcal{X}}\left(x^k - \beta h_f^k\right) \tag{2b}$$

where h_g^k is an unbiased stochastic gradient estimator of $\mathbb{E}_\phi[g(x^k, y^k; \phi)]$, h_f^k is a (possibly biased) stochastic gradient estimator of $F(x^k)$, and, α and β are stepsizes. An immediate challenge in analyzing the projected SGD updates in (2) is that h_f^k is usually biased due to the inaccessibility of $y^*(x)$. Moreover, the bias is roughly proportional to the LL accuracy $\|y^{k+1} - y^*(x^k)\|$, but the latter is not ensured to be small enough after $\mathcal{O}(1)$ LL steps. Therefore, if we directly apply the existing analysis for nonconvex constrained single level problem (Davis and Drusvyatskiy, 2019) to even merely UL constrained bilevel problem, it either leads to suboptimal rate (Hong et al., 2020) or requires additional LL corrections (Chen et al., 2022a), let alone coupled with LL constraints. Leveraging the smoothness of the projection to linear equality constraints, we develop problem-specific proof and establish the convergence of alternating projected SGD comparable to the unconstrained case.

Furthermore, alternating projected SGD in (2) may not be suitable for the scenarios where evaluating projections is expensive since it calls projections at every step. For example, when the constraint set represents the consensus constraint in federated bilevel learning (Tarzanagh et al., 2022), projection amounts to averaging the gradients of all clients (Parikh et al., 2014), which suffers from high communication cost. This motivates a projection-efficient algorithm for (1) beyond alternating projected SGD.

1.1 Contributions

In this context, we consider the bilevel optimization with (possibly coupled) equality constraints. We analyze the convergence rate for alternating projected SGD, AiPOD for short, propose a projection efficient variant of AiPOD and apply it to the federated bilevel optimization. We summarize our contributions as follows.

C1) We provide the first nonasymptotic analysis of AiPOD for bilevel optimization with both UL and LL constraints and attain the $\tilde{\mathcal{O}}(\epsilon^{-2})$ sample complexity to achieve ϵ stationary point of (1), which matches the complexity of alternating SGD algorithm for unconstrained bilevel problem (Chen et al., 2021).

C2) We propose an efficient variant of AiPOD termed E-AiPOD tailored to the setting where evaluating projection is costly, and establish an improved projection complexity of the UL and LL variable over AiPOD respectively, i.e. from $\tilde{\mathcal{O}}(\epsilon^{-2})$ to $\tilde{\mathcal{O}}(\epsilon^{-2}/T)$ and $\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$, where T is the UL projection interval.

C3) We show the implication of the proposed method on federated bilevel learning and provide the enhanced communication complexity over the state-of-the-art work (Tarzanagh et al., 2022). Experiments on numerical examples and federated hyperparameter optimization are provided to verify our theoretical findings.

1.2 Technical challenges

We highlight the technical challenges for the analysis.

T1) The state-of-the-art analysis of unconstrained bilevel optimization (Ghadimi and Wang, 2018; Hong et al., 2020; Chen et al., 2021) relies on the smoothness of the implicit gradient mapping $\nabla y^*(x)$. However, the well-known formula of $\nabla y^*(x)$ does not hold when LL problem has constraints so that the smoothness of $y^*(x)$ is unknown.

T2) The update of UL can be viewed as biased projected SGD. However, the bias of gradient estimator will lead to suboptimal rates if we directly apply the general analysis of projected SGD (Davis and Drusvyatskiy, 2019) to the UL sequence x, since we can not separate out a negative term to mitigate the LL bias.

T3) The Lyapunov function that is critical in analyzing the single-loop unconstrained bilevel optimization (Chen et al., 2021) is insufficient for the analysis of our projection-efficient variant E-AiPOD due to the additional errors caused by skipping projections steps.

1.3 Related works

To put our work in context, we review prior art from the following two categories.

Unconstrained bilevel optimization. Bilevel optimization has a long history back to (Bracken and McGill, 1973) and has inspired a rich literature, e.g., (Ye and Zhu, 1995; Vicente and Calamai, 1994; Colson et al., 2007; Sinha et al., 2017). Later on, spurred by the advancement of hyperparameter optimization (Maclaurin et al., 2015; Franceschi et al., 2018) and meta learning (Finn et al., 2017), bilevel optimization has received more attention as a unified tool for problems with nested structure. With the more use cases in large-scale machine learning, developing stochastic methods with finite-time guarentee has become the recent focus in the

	AiPOD	E-AiPOD	BSA	ALSET	stocBiO	FSLA	TTSA	IG-AL
stochasticity	✓	✓	✓	✓	✓	✓	✓	✗
UL constraint	✓	✓	✗	✗	✗	✗	✓	✗
LL constraint	✓	✓	✗	✗	✗	✗	✗	✓
y-update	projected SGD	proxskip	SGD	SGD	SGD	SGD	SGD	ALM
sample complexity	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\mathcal{O}(\epsilon^{-3})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\mathcal{O}(\epsilon^{-2.5})$	/
UL projection	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-2}/T)$	/	/	/	/	$\mathcal{O}(\epsilon^{-2.5})$	/
LL projection	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$	/	/	/	/	/	/

Table 1: Sample complexity and projection complexity of our methods AiPOD, E-AiPOD and the state-of-the-art works on bilevel problem (BSA in (Ghadimi and Wang, 2018), ALSET in (Chen et al., 2021), stoBiO in (Ji et al., 2021), FSLA in (Li et al., 2022a), TTSA in (Hong et al., 2020)), IG-AL in (Tsaknakis et al., 2022)) to achieve an ϵ stationary point, where proxskip is in (Mishchenko et al., 2022) and ALM denotes Augmented Lagrangian methods. The notation $\tilde{\mathcal{O}}$ omits the polynomial dependency on $\log(\epsilon^{-1})$ terms.

area of bilevel optimization. The interest of the nonasympototic analysis of the stochastic bilevel optimization has been stimulated since a recent work (Ghadimi and Wang, 2018) that tackles the bilevel setting where the LL objective is strongly convex and Lipschitz smooth. As $\nabla F(x)$ contains the Hessian inverse of the LL objective which is computational expensive, it has emerged various numeric approximation methods including Neumann series approximation (Ghadimi and Wang, 2018), unrolling differentiation (Grazzi et al., 2020), and conjugate gradient (Ji et al., 2021); see a comparison in (Lorraine et al., 2020; Ji et al., 2022). In terms of the alternating SGD algorithm, (Chen et al., 2021) achieved the $\tilde{\mathcal{O}}(\epsilon^{-2})$ sample complexity, which matches the results for single level case. Recently, (Li et al., 2022a) has put forward a fully single-loop algorithm which updates the Hessian inverse approximation dynamically. Beyond the alternating SGD framework, (Khanduri et al., 2021; Yang et al., 2021b) incorporate variance reduction techniques to further accelerate the convergence; see a recent survey for bilevel optimization (Liu et al., 2021a). Nevertheless, none of them solve (1) with both UL and LL constraints.

Constrained bilevel optimization. While the nonasympotic convergence for various approaches in unconstrained bilevel setting has been extensively studied in literature, the nonasymptotic analysis of stochastic algorithms for constrained bilevel optimization problems is very limited. Some recent efforts have been devoted to tackle the constrained UL setting. Hong et al. (2020) has established $\mathcal{O}(\epsilon^{-2.5})$ rate of TTSA which applied SGD in LL update and projected SGD in UL update; Chen et al. (2022a) has achieved $\tilde{\mathcal{O}}(\epsilon^{-2})$ convergence rate by adding additional corrections on LL update for the stochastic setting; Chen et al. (2022c) has proved the $\mathcal{O}(\epsilon^{-1})$ convergence rate of proximal accelerated gradient based method for deterministic constrained UL problem under the Kurdyka-Łojasiewicz geometry. As for constrained LL problem, the vast majority of works focus on either the asymptotic analysis, e.g., initialization auxiliary method (Liu et al., 2021b), value function based approach (Gao et al., 2022b); or design aspects, e.g. optimality of bilevel problem (Dempe et al., 2007; Ye and Zhu, 2010), reformulation (Dempe and Zemkoho, 2013; Brotcorne et al., 2013), and differential properties (Gould et al., 2016; Dyro et al., 2022). The notable exception is a recent breakthrough (Tsaknakis et al., 2022), which tackles linearly inequality constrained LL problem in a double-loop manner, i.e. update UL variable after attaining a sufficiently accurate LL solution. However, the overall iteration/sample complexity has not been established therein.

We summarize the comparison of our work with the closely related prior art in Table 1.

2 AiPOD for Stochastic Bilevel Problems

In this section, we introduce notations, present the AiPOD algorithm and establish the finite-time convergence for it.

2.1 Preliminaries

For convenience, we define $g(x,y) := \mathbb{E}_\phi\left[g(x,y;\phi)\right]$ and $f(x,y) := \mathbb{E}_\xi\left[f(x,y;\xi)\right]$. We also define $\nabla_{yy} g(x,y)$ as the Hessian of g with respsect to y and denote

$$\nabla_{xy} g(x,y) = \begin{bmatrix} \frac{\partial^2}{\partial x_1 \partial y_1} g(x,y) & \cdots & \frac{\partial^2}{\partial x_1 \partial y_{d_y}} g(x,y) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial x_{d_x} \partial y_1} g(x,y) & \cdots & \frac{\partial^2}{\partial x_{d_x} \partial y_{d_y}} g(x,y). \end{bmatrix}$$

We use $\|\cdot\|$ to denote the ℓ_2 norm for vectors and Frobenius norm for matrix. We also denote $A^\dagger$ and $B^\dagger$ as the the Moore Penrose inverse of A and B (James, 1978). Moreover, we define $P_x := I - B^\dagger B$ as the projection matrix over the null space of x and $\|x\|_{P_x} := \sqrt{x^\top P_x x}$ as the P_x weighted Euclidean norm.

In literature, the common convergence metric for constrained optimization is (Ghadimi et al., 2016)

$$\mathbb{E}[\|\lambda^{-1}(x - \mathrm{Proj}_{\mathcal{X}}(x - \lambda \nabla F(x)))\|^2] \tag{3}$$

for some $\lambda > 0$. In (1), since $\mathcal{X}$ contains only linear equality constraints, (3) can be simplified according to the following lemma, the proof of which will be deferred to Appendix B.

Lemma 1. *For any $x \in \mathcal{X}$ and any $\lambda > 0$, we have that*

$$\|\lambda^{-1}(x - \mathrm{Proj}_{\mathcal{X}}(x - \lambda \nabla F(x)))\|^2 = \|\nabla F(x)\|^2_{P_x}.$$

Therefore, we define the ϵ stationary point x for (1) as

$$\mathbb{E}\left[\|\nabla F(x)\|^2_{P_x}\right] \le \epsilon. \tag{4}$$

If $B = 0$, (4) is reduced to $\mathbb{E}\left[\|\nabla F(x)\|^2\right] \le \epsilon$, which is the standard stationary measure for unconstrained stochastic bilevel optimization settings (Ghadimi and Wang, 2018; Ji et al., 2021; Chen et al., 2021).

2.2 The basic algorithm

In this section, we will introduce the basic version of AiPOD algorithm for (1), which updates x and y in an alternating projected SGD manner.

At a given UL iteration k, we update y^{k+1} by the output of the S projected SGD steps for $g(x^k, y)$. With initialization $y^{k,0} = y^k$, we update

$$y^{k,s+1} = \mathrm{Proj}_{\mathcal{Y}(x^k)}\left(y^{k,s} - \beta \nabla g(x^k, y^{k,s}; \phi^{k,s})\right) \tag{5}$$

and set $y^{k+1} = y^{k,S}$. For UL, the gradient $\nabla F(x)$ can be calculated by the chain rule

$$\nabla F(x) = \nabla_x f(x, y^*(x)) + \nabla^\top y^*(x) \nabla_y f(x, y^*(x)) \tag{6}$$

where the implicit mapping $\nabla y^*(x)$ is essential.

Without LL constraint, $\nabla y^*(x)$ can be derived from the LL optimality condition $\nabla_y g(x, y^*(x)) = 0$ as (Ghadimi and Wang, 2018)

$$\nabla y^*(x) = -\nabla^{-1}_{yy} g(x, y^*(x)) \nabla_{yx} g(x, y^*(x)). \tag{7}$$

We generalize it to the constrained LL setting, and establish the implicit gradient for constrained LL in the following lemma, the proof of which is deferred to Appendix C.1.

Lemma 2. *Define V_2 as the orthogonal basis of $\mathrm{Ker}(A) := \{y \mid Ay = 0\}$. When $g(x, y)$ is twice differentiable and strongly convex over y, the implicit gradient $\nabla y^*(x)$ can be written as*

$$\begin{aligned}\nabla y^*(x) = &- \underbrace{V_2 (V_2^\top \nabla_{yy} g(x, y^*(x)) V_2)^{-1} V_2^\top}_{P_1} \\ &\times \underbrace{\left(\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\right) - A^\dagger \nabla h(x)}_{P_2}\end{aligned} \tag{8}$$

where $A^\dagger$ is the Moore-Penrose inverse of A.

Compared with (7), the term P_1 in (8) can be viewed as projecting $\nabla^{-1}_{yy} g(x, y^*(x))$ to $\mathrm{Ker}(A)$; while P_2 accounts for the coupling constraints $Ay + h(x) = c$.

Algorithm 1 AiPOD for constrained bilevel problem

Initialization: x^0, y^0, stepsizes $\{\alpha, \beta\}$, N
for $k = 0$ **to** $K - 1$ **do**
  for $s = 0$ **to** $S - 1$ **do** ▷ Set $y^{k,0} = y^k$
    update $y^{k,s+1}$ by (5).
  end for ▷ Set $y^{k+1} = y^{k,S}$
  evaluate w^k in (9b)
  calculate h^k_f in (9a)
  update $x^{k+1} = \mathrm{Proj}_{\mathcal{X}}(x^k - \alpha h^k_f)$
end for

With the similar spirits of the existing works (Ghadimi and Wang, 2018; Hong et al., 2020; Chen et al., 2021), we obtain the UL gradient estimator h^k_f at UL iteration k by setting $x = x^k$, approximating $y^*(x^k)$ by y^{k+1} in (6) and estimating P_1 by Neumann series. Thus, h^k_f is defined as

$$h^k_f := \nabla_x f(x^k, y^{k+1}; \xi^k) + w^k \tag{9a}$$

where w^k is defined as (cf. (6) and (8))

$$\begin{aligned} w^k := &\Big(\nabla h(x^k)^\top A^{\dagger\top} \nabla_{yy} g(x^k, y^{k+1}; \phi^k_{(0)}) \\ &- \nabla_{xy} g(x^k, y^{k+1}; \phi^k_{(0)})\Big) \\ &\times V_2 \left[\frac{\tilde{c} N}{\ell_{g,1}} \prod_{n=1}^{N'} \left(I - \frac{\tilde{c}}{\ell_{g,1}} V_2^\top \nabla_{yy} g\left(x^k, y^{k+1}; \phi^k_{(n)}\right) V_2\right)\right] V_2^\top \\ &\times \nabla_y f(x^k, y^{k+1}; \xi^k) - \nabla h(x^k)^\top A^{\dagger\top} \nabla_y f(x^k, y^{k+1}; \xi^k) \end{aligned} \tag{9b}$$

where $\tilde{c} \in (0, 1]$ is a given constant, N' is drawn uniformly at random from $\{0, \cdots N-1\}$, and $\{\phi^k_{(0)}, \cdots, \phi^k_{(N')}\}$ are i.i.d samples.

We can update x^{k+1} by projected SGD with estimator h^k_f in (9a) in the outer loop and update y^{k+1} by projected SGD in the inner loop (5); see the full algorithm in Algorithm 1.

2.3 Theoretical analysis

For the analysis, we make the following assumptions.

Assumption 1. *Assume that $f, \nabla f, \nabla g, \nabla_{xy} g, \nabla_{yy} g, h$ and ∇h are Lipschitz continuous with $\ell_{f,0}, \ell_{f,1}, \ell_{g,1}, \ell_{g,2}, \ell_{g,2}, \ell_{h,0}, \ell_{h,1}$, respectively.*

Assumption 2. *For any fixed x, assume that $g(x, y)$ is μ_g-strongly convex with respect to $y \in \mathbb{R}^{d_y}$.*

Assumption 3. *The stochastic estimators $\nabla f(x, y; \xi)$, $\nabla g(x, y; \phi), \nabla_{xy} g(x, y; \phi)$ and $\nabla_{yy} g(x, y; \phi)$ are unbiased estimators of $\nabla f(x, y), \nabla g(x, y), \nabla_{xy} g(x, y)$ and $\nabla_{yy} g(x, y)$, and their variance are bounded by $\sigma^2_f, \sigma^2_{g,1}, \sigma^2_{g,2}$ and $\sigma^2_{g,2}$, respectively.*

Assumption 4. *The set $\mathcal{X}$ is nonempty. For any x, the set $\mathcal{Y}(x)$ is nonempty.*

Assumption 1–3 are standard for stochastic bilevel optimization (Ghadimi and Wang, 2018; Hong et al., 2020; Ji et al.,

2021; Khanduri et al., 2021; Chen et al., 2021, 2022a; Li et al., 2022a). Assumption 4 is to ensure the feasibility of the problem (1). Since $Ay = b$ has solution y if and only if $AA^\dagger b = b$ according to (James, 1978), Assumption 4 is equivalent to $AA^\dagger(c - h(x)) = c - h(x)$. One sufficient but not necessary condition for Assumption 4 is that A is full row rank, which does not impose any additional requirement on $h(x)$. Another sufficient condition is $\forall x$, $c - h(x) \in \mathrm{Ran}(A)$, e.g., $h(x) = 0, c = 0$, where A is not necessarily full row rank.

One of the keys to establishing $\tilde{\mathcal{O}}(\epsilon^{-2})$ convergence rate of unconstrained bilevel optimization (Chen et al., 2021; Li et al., 2022a) is to utilize the smoothness of $y^*(x)$. Thanks to the singular value decomposition, we can obtain the smoothness of $y^*(x)$ for linearly equality constrained LL (1) in the next lemma, the proof of which is deferred to Appendix C.1.

Lemma 3. *Under Assumption 1–2 and 4, $y^*(x)$ is L_y-Lipschitz continuous and L_{yx}- smooth, where the constants L_y and L_{yx} are specified in Appendix C.1.*

However, due to the UL constraint, the proof for unconstrained stochastic bilevel optimization (Chen et al., 2021) cannot be applied even with the smoothness of $y^*(x)$ in Lemma 3. For constrained UL with LL unconstrained bilevel problem, the recent works (Hong et al., 2020; Chen et al., 2022a) have leveraged the Moreau envelope technique in (Davis and Drusvyatskiy, 2019) to develop projected implicit SGD. However, when the stochastic gradient estimator is biased, the bias term delays at a slower timescale. As a result, existing methods either suffer from the suboptimal $\mathcal{O}(\epsilon^{-2.5})$ iteration complexity (Hong et al., 2020), or require an additional correction in the LL update to achieve the $\mathcal{O}(\epsilon^{-2})$ iteration complexity (Chen et al., 2022a). Owing to the special property of linear-equality constraits in Lemma 1, we establish the $\mathcal{O}(\epsilon^{-2})$ iteration complexity of Algorithm 1 in the next lemma without resorting to Moreau envelope technique. The proof is deferred to Appendix C.4.

Theorem 4 (Convergence of AiPOD). *Under Assumption 1–4, if we choose $N = \mathcal{O}(\log K)$ and*

$$\alpha = \min\left(\bar{\alpha}_1, \bar{\alpha}_2, \frac{\bar{\alpha}}{\sqrt{K}}\right), \quad \beta = \frac{5L_f L_y + \eta L_{yx}\tilde{C}_f^2}{\mu_g}\alpha$$

where $\bar{\alpha}_1, \bar{\alpha}_2$ are defined in (69) *then for any $S = \mathcal{O}(1)$ in Algorithm 1, we have*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] = \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{K}}\right).$$

Theorem 4 shows that Algorithm 1 achieves an ϵ- stationary point by $\tilde{\mathcal{O}}(\epsilon^{-2})$ iterations, which matches the iteration complexity of single level stochastic projected gradient descent method (Davis and Drusvyatskiy, 2019) and the unconstrained bilevel SGD method (Chen et al., 2021).

3 Extensions to Lazy Projections

In this section, we focus on the case when evaluating projection is expensive and then propose a projection efficient method E-AiPOD to avoid frequent projection steps.

Besides the explicit projections for x- and y- updates in Algorithm 1, calculating w^k in (9b) also requires projecting $\nabla_{yy}g(x^k, y^{k+1}; \phi_{(n)}^k)$ onto the null space of the LL problem, i.e. calculating $V_2^\top \nabla_{yy}g(x^k, y^{k+1}; \phi_{(n)}^k)V_2$. We use the following ways to save projections.

Lazy LL projection. We leverage a recent breakthrough of projected SGD called Proxskip (Mishchenko et al., 2022) in the LL update which evaluates projection lazily with probability $0 < p < 1$. At each LL iteration s, we first perform an SGD update corrected by the residual $r^{k,s}$ as

$$\hat{y}^{k,s+1} = y^{k,s} - \beta(\nabla_y g(x^k, y^{k,s}; \phi^{k,s}) - r^{k,s}). \tag{10a}$$

With probability $1 - p$, we skip the projection and keep $y^{k,s+1} = \hat{y}^{k,s+1}, r^{k,s+1} = r^{k,s}$; with probability p, we update LL parameter $y^{k,s}$ and the residual $r^{k,s}$ as

$$y^{k,s+1} = \mathrm{Proj}_{\mathcal{Y}(x^k)}\left(\hat{y}^{k,s+1} - \beta r^{k,s}/p\right) \tag{10b}$$

$$r^{k,s+1} = r^{k,s} + p(y^{k,s+1} - \hat{y}^{k,s+1})/\beta. \tag{10c}$$

In (10a), $r^{k,s}$ compensates the error of lazy projection and is updated every $1/p$ rounds in expectation so that the corrected gradient descent in (10a) can approximate the projected SGD update in (5). If the overall convergence rate for AiPOD equipped with LL lazy projection mechanism does not deteriorate, the expected number of projection evaluations is reduced from KS to pKS.

Delay computation of w^k and reduce UL projection. To save UL projection, at upper iteration k, we calculate w^k by (9b) once and update x^k by SGD T times, given by

$$x^{k,t+1} = x^{k,t} - \alpha h_f^{k,t}, \ \ t = 0, \cdots, T-1, \quad \text{with } x^{k,0} = x^k \tag{11a}$$

where the estimator $h_f^{k,t}$ is defined below

$$h_f^{k,t} := \nabla_x f(x^{k,t}, y^{k+1}; \xi^{k,t}) + w^k. \tag{11b}$$

where w^k is defined in (9b). Compared with h_f^k in (9a), $h_f^{k,t}$ can be regarded as the UL gradient estimator at $x^{k,t}$ obtained by delayed Hessian inverse vector approximation w^k. After T rounds, we update x^{k+1} by

$$x^{k+1} = (1 - \delta)x^k + \delta\,\mathrm{Proj}_{\mathcal{X}}(x^{k,T}) \tag{12}$$

where $\delta \geq 1$ positively correlates to T so that scales the projected descent stepsizes. In this way, we only project the Hessian estimator to the null space to evaluate w^k at $t = 0$

Algorithm 2 E-AiPOD for constrained bilevel problem

Initialize: x^0, y^0, stepsizes $\{\alpha, \beta, \delta\}$, probability p, N
for $k = 0$ **to** $K-1$ **do**
  for $s = 0$ **to** $S-1$ **do** ▷ set $y^{k,0} = y^k$; $r^{k,0} = r^k$
    update $\hat{y}^{k,s+1}$ by (10a)
    draw Bernoulli $\theta^{k,s}$ with p
    if $\theta^{k,s} = 1$ **then**
      update $y^{k,s+1}$ by (10b)
    else
      $y^{k,s+1} = \hat{y}^{k,s+1}$
    end if
    update $r^{k,s+1}$ by (10c)
  end for ▷ set $r^{k+1} = r^{k,S}$ and $y^{k+1} = y^{k,S}$
  compute w^k defined in (9b).
  for $t = 0$ **to** $T-1$ **do**
    calculate $h_f^{k,t}$ in (11b)
    update $x^{k,t+1}$ by (11a)
  end for ▷ set $x^{k,0} = x^k$
  update x^{k+1} by (12)
end for

and project x sequence at the end of the T-loop. The error resulting from the delayed w^k will be shown to be bounded by $\mathcal{O}(\alpha^2)$, which is not the dominating term in the analysis.

We summarize E-AiPOD in Algorithm 2 and characterize its convergence rate by using a new Lyapunov function as

$$\mathbb{V}^k := F(x^k) + \frac{L_f}{L_r}\left(\|y^*(x^k) - y^k\|^2 + \frac{\beta^2}{p^2}\|r^k - r^*(x^k)\|^2\right) \quad (13)$$

where L_f and L_r are constants defined in Appendix C.1 and C.2. The complexity bound of E-AiPOD is stated in the following theorem.

Theorem 5 (Convergence of E-AiPOD). *Under Assumption 1–4, if we choose stepsizes such that $\alpha\delta T < \bar{\alpha}$, where $\bar{\alpha} < 1$ is a constant formally defined in (94), and let $\beta = \mathcal{O}(\alpha T), p = \mathcal{O}(\sqrt{\beta}), S = \mathcal{O}(1)$, then the sequences generated by Algorithm 2 satisfies*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] \le \frac{2(\mathbb{V}^0 - F^*)}{\alpha\delta TK} + 2c_1\sigma_{g,1}^2\alpha\delta T + 2c_2\tilde{\sigma}_f^2\alpha\delta + \mathcal{O}(\alpha^2\delta^2 T) \quad (14)$$

where c_1 and c_2 are constant formally defined in Appendix D.3 and F^* is the lower bound of $F(x)$.

With proper choices of α and δ, the four terms in (14) could vanish simultaneously. The following corollary shows the results for vanilla periodical projections when $\delta = 1$.

Corollary 6 (Reduction of LL projection). *Under the same condition of Theorem 5, if we choose $\alpha = \frac{\bar{\alpha}}{T\sqrt{K}}$, $\delta = 1$, the convergence rate of Algorithm 2 is*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] = \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{K}}\right). \quad (15)$$

Since $K = \tilde{\mathcal{O}}(\epsilon^{-2})$ and $p = \mathcal{O}(\sqrt{\beta}) = \mathcal{O}(K^{-\frac{1}{4}})$, the total number of evaluations of LL projection is reduced to

$$\mathcal{O}(pK) = \mathcal{O}(K^{\frac{3}{4}}) = \tilde{\mathcal{O}}(\epsilon^{-1.5}). \quad (16)$$

Corollary 6 implies that the convergence rate of E-AiPOD is the same as that of AiPOD when $\delta = 1$, but the LL projection complexity can be reduced to $\tilde{\mathcal{O}}(\epsilon^{-1.5})$. Compared with Proxskip (Mishchenko et al., 2022) which improves the projection complexity on $\kappa = \ell_{g,1}/\mu_g$, we can further achieve the reduction on ϵ owing to the smaller LL stepsize $\beta = \mathcal{O}(1/\sqrt{K})$. Besides, the next corollary shows the benefit for enlarging δ, which further reduces the iteration and projection complexity of E-AiPOD.

Corollary 7 (Reduction of UL projection). *Under the same condition of Theorem 5, if we choose $\alpha = \frac{\bar{\alpha}}{T\sqrt{K}}$, $\delta = \sqrt{T}$, and select LL batch size as $\mathcal{O}(T)$ such that $\sigma_g^2 = \mathcal{O}(1/T)$, the convergence rate of E-AiPOD is*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] = \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{TK}}\right). \quad (17)$$

As a result, the sample complexity for both the UL and LL is $TK = \tilde{\mathcal{O}}(\epsilon^{-2})$; the total number of the UL projections reduces to $\mathcal{O}(K) = \tilde{\mathcal{O}}(\epsilon^{-2}/T)$; and, the total number of LL projections reduces to $\mathcal{O}(pK) = \mathcal{O}(K^{\frac{3}{4}}) = \tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$.

Corollary 7 implies with larger δ, increasing T can accelerate the convergence rate and improve the projection complexity without degrading the sample complexity for both levels. Compared with the single level case, σ_g^2 can be seen as the additional error caused by the LL stochasticity. The idea behind reducing the UL projection complexity is to reduce the variance of the averaged gradient estimator by T gradient descent steps and use larger δ to balance the projection stepsize. However, this can not reduce the variance of LL; see the different terms for $\tilde{\sigma}_f^2$ and σ_g^2 over T in (14). Therefore, we need to increase the LL batch size correspondingly to reduce σ_g^2 in bilevel problem. By virtual of the faster rate, the sample complexity for LL is still $\tilde{\mathcal{O}}(\epsilon^{-2})$.

4 Application to Federated Bilevel Learning

We consider the federated bilevel optimization (Tarzanagh et al., 2022) in the consensus form

$$\min_{x\in\mathcal{X}} \; F(x) = \frac{1}{M}\sum_{m=1}^{M} f_m\left(x_m, y_m^*(x_m)\right)$$
$$\text{s.t.} \quad y^*(x) = \arg\min_{y\in\mathcal{Y}} \frac{1}{M}\sum_{m=1}^{M} g_m(x_m, y_m) \quad (18)$$

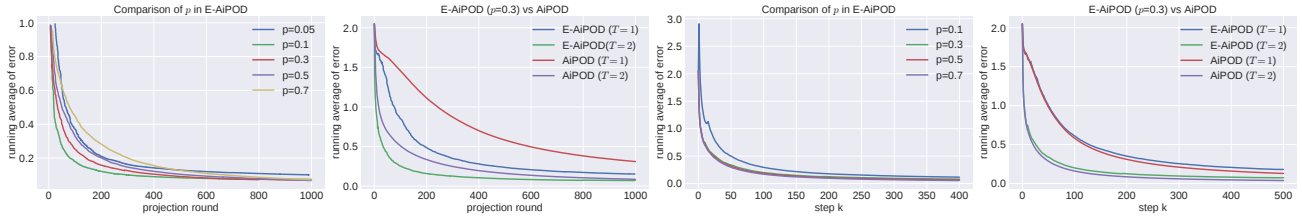

Figure 1: Impact of p in E-AiPOD (left) and comparison between AiPOD and E-AiPOD (right). Here the running average of *error* is defined as $\frac{1}{K}\sum_{k=1}^{K}(\|\nabla F(x^k)\|^2_{P_x} + \|y^{k+1} - y^*(x^k)\|^2)$.

where each client $m \in [M] := \{1, \cdots M\}$ maintains its local model x_m, y_m and is only accessible to its individual function (f_m, g_m); $x = [x_1, \cdots, x_M]^\top$ and $y = [y_1, \cdots, y_M]^\top$ are the collection of individual models; $y^*(x) = [y_1^*(x_1), \cdots, y_M^*(x_M)]^\top$ is the optimal LL model; and $\mathcal{X} = \{x \mid x_1 = \cdots = x_M\}$ and $\mathcal{Y} = \{y \mid y_1 = \cdots = y_M\}$ denote the consensus set.

With $\mathbf{I}_d \in \mathbb{R}^{d\times d}$ denoting as the identity matrix and $\mathbf{1}_M \in \mathbb{R}^M$ denoting as the all-1 vector, we can define the consensus matrix A and calculate the orthogonal basis of its kernel as

$$A := \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \otimes \mathbf{I}_d, \quad V_2 := \frac{\mathbf{1}_M}{\sqrt{M}} \otimes \mathbf{I}_d \quad (19)$$

where $\otimes$ is the Kronecker product, $A \in \mathbb{R}^{d(M-1)\times dM}$ and $V_2 \in \mathbb{R}^{dM\times d}$. We can define B the same as A except for dimension d. In the federated bilevel setting, with $e = c = h(x) = 0$, the UL and LL constraint sets become $\mathcal{X} = \{x \mid Bx = 0\}$ and $\mathcal{Y} = \{y \mid Ay = 0\}$.

Therefore, the UL gradient in (6) can be specialized as $\nabla F(x) = [\nabla_{x_1} F(x), \cdots, \nabla_{x_M} F(x)]^\top$ in the federated bilevel setting (18), where

$$\nabla_{x_m} F(x) = \nabla_{x_m} f_m(x_m, y_m^*(x_m)) + \frac{\nabla y_m^{*\top}(x_m)}{M} \sum_{m=1}^{M} \nabla_{y_m} f_m(x_m, y_m^*(x_m)) \quad (20a)$$

$$\text{with } \nabla y_m^*(x_m) = -\left(\frac{1}{M}\sum_{m=1}^{M} \nabla_{yy} g_m(x_m, y_m^*(x_m))\right)^{-1} \times \nabla_{yx} g_m\left(x_m, y_m^*(x_m)\right). \quad (20b)$$

Moreover, evaluating the projections in federated bilevel learning is equivalent to averaging x_m and y_m, i.e.,

$$\text{Proj}_{\mathcal{X}}(x) = (\bar{x}, \cdots, \bar{x}), \quad \text{with } \bar{x} = \frac{1}{M}\sum_{m=1}^{M} x_m \quad (21a)$$

$$\text{Proj}_{\mathcal{Y}}(y) = (\bar{y}, \cdots, \bar{y}), \quad \text{with } \bar{y} = \frac{1}{M}\sum_{m=1}^{M} y_m. \quad (21b)$$

With the above facts, we are able to apply E-AiPOD to the federated bilevel setting, which is summarized in Algorithm 3 in Appendix.

Besides, the convergence of Algorithm 3 is inherited from the result of E-AiPOD in Theorem 5 and the weighted norm measure (4) in our analysis coincides with the measure in non-consensus form of federated bilevel learning (Tarzanagh et al., 2022) (see the proof in Appendix E.2), so we have the following corollaries and comparisons.

Corollary 8 (Convergence rate and sample complexity). *Under the same condition of Theorem 5, the convergence rate of Algorithm 3 is*

$$\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}\left[\left\|\nabla F(x^k)\right\|^2_{P_x}\right] = \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{TK}}\right).$$

Therefore, the sample complexity of Algorithm 3 for both the UL and LL is $TK = \tilde{\mathcal{O}}(\epsilon^{-2})$. Next we will leverage Corollary 7 to establish communication complexity.

Corollary 9 (Communication complexity). *Under the same condition and parameter choices with Corollary 7, the UL communication is reduced to $\mathcal{O}(K) = \tilde{\mathcal{O}}(\epsilon^{-2}/T)$, while the LL communication is reduced to*

$$\mathcal{O}(pK) = \mathcal{O}(K^{\frac{3}{4}}) = \tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}}).$$

Compared with the state-of-the-art work FedNest (Tarzanagh et al., 2022), the sample complexity of FedNest and Algorithm 3 are the same, while the UL and LL communication complexity of Algorithm 3 are reduced from $\tilde{\mathcal{O}}(\epsilon^{-2})$ to $\tilde{\mathcal{O}}(\epsilon^{-2}/T)$ and $\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$, respectively. It is worthy to mention that we simplify the local updates from SVRG-type in FedNest to SGD-type in Algorithm 3 and our communication gains do not require transmitting additional messages besides local models in the communication rounds (e.g., momentum variables in (Li et al., 2022b)). Compared with Scaffnew (Mishchenko et al., 2022), we can further reduce the LL communication complexity on ϵ since we can tolerate $\mathcal{O}(1/\sqrt{K})$ LL stepsize. Due to the space limitation, we will defer the literature review of federated bilevel learning in Appendix E.3 and provide a comparison of communication complexity in in Appendix E.4.

5 Experiments

To validate the theoretical results and evaluate the empirical performance of our methods, we conduct experiments in

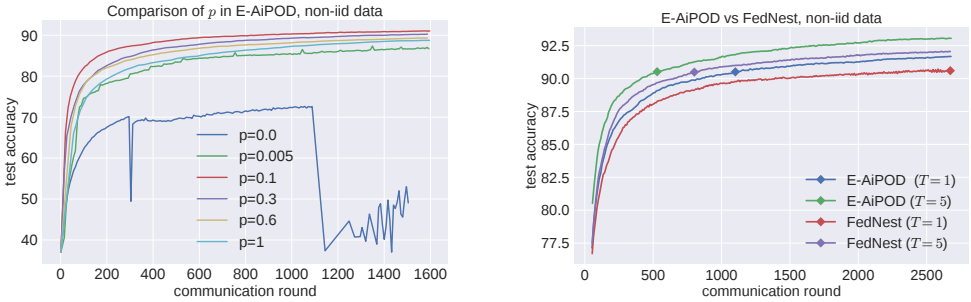

Figure 2: Federated hyper-representation learning: Impact of communication probability p (left) and comparison of our algorithm with FedNest (right). The experiments are run on the MNIST dataset with non-i.i.d. distribution.

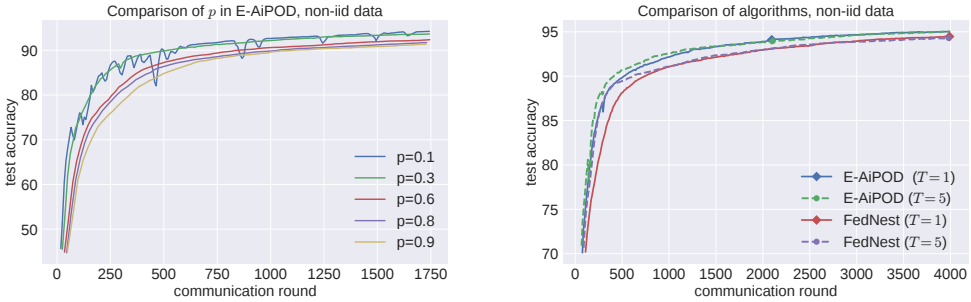

Figure 3: Federated learning from imbalanced data: Impact of communication probability p (left) and comparison of our algorithm with FedNest (right). The experiments are run on an imbalanced MNIST with non-i.i.d. distribution.

both synthetic tests and practical federated bilevel learning tasks, which will be presented in this section.

Synthetic experiments. We first consider a special case of the coupling equality-constrained bilevel problem

$$\min_{x\in\mathcal{X}} \ \sin\left(c^\top x + d^\top y^*(x)\right) + \ln\left(\|x + y^*(x)\|^2 + 1\right)$$
$$\text{s.t.} \ \ y^*(x) = \arg\min_{y\in\mathcal{Y}(x)} \frac{1}{2}\|x - y\|^2$$

where $\mathcal{X} = \{x \mid Bx = 0\} \subset \mathbb{R}^{100}$, $\mathcal{Y}(x) = \{y \mid Ay + Hx = 0\} \subset \mathbb{R}^{100}$, and A, B, H, c, d are randomly generated non-zero matrices or vectors. To guarantee that $\mathcal{Y}(x)$ and $\mathcal{X}$ are not singleton, the matrixes A and B are rank-deficient matrices. In the simulation, we use the noisy gradient where the Gaussian noise with zero mean and the standard deviation of 0.1 is added. In this setting, Assumptions 1–4 are satisfied.

The test results are reported in Figure 1. In the left figure, we test the impact of the probability p on the projection complexity and the iteration complexity. It can be observed that E-AiPOD with relatively small p has almost the same iteration complexity (as indicated in the lower left figure) while it significantly saves projection rounds (see upper left figure). We also compare AiPOD and E-AiPOD in the right figure. It can be observed that E-AiPOD is able to save projection while maintaining the same iteration complexity as that of AiPOD, which is aligned with our theory.

Federated representation learning. With additional details provided in Appendix F.2, in this part, we apply E-AiPOD in Algorithm 2 to the federated representation learning task, which builds on the bilevel representation learning (Franceschi et al., 2018). The goal is to learn a joint representation model and a client-specific header while protecting data privacy. The experimental results are reported in Figure 2. From the left figure of Figure 2, a relatively small value of p helps save communication rounds while a too small value of p might degrade performance. With a properly chosen $p = 0.1$, it can be observed from the right figure that E-AiPOD outperforms FedNest (Tarzanagh et al., 2022) in terms of communication complexity.

Federated loss function tuning. With additional details provided in Appendix F.3, in this part, we apply E-AiPOD to the federated learning from imbalanced data task. The goal is to learn a good model that guarantees both the fairness and generalization from datasets with under-represented classes (Li et al., 2021). The experimental results are reported in Figure 3. From the left figure of Figure 3, a relatively small value of p helps save communication rounds. With $p = 0.3$, it can be observed from the right figure that E-AiPOD outperforms FedNest in terms of communication complexity. Algorithms with a larger T have faster convergence at the start, but achieve the same accuracy as those with $T = 1$.

6 Conclusions

In this paper, we established the first finite-time convergence of alternating projected SGD algorithm (AiPOD) for equality-constrained bilevel problems which matches the state-of-the-art result for alternating SGD in unconstrained bilevel problem and also the single-level projected SGD. We proposed a projection-efficient variant E-AiPOD for the settings where evaluating projection is costly. E-AiPOD can reduce the UL and LL projection complexity to $\tilde{\mathcal{O}}(\epsilon^{-2}/T)$ and $\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$, respectively. We applied E-AiPOD to federated bilevel setting and achieved the reduction in communication complexity. Experiments verified our theoretical results and demonstrated the effectiveness of our methods.

Acknowledgment

The work of Q. Xiao, H. Shen and T. Chen was supported by National Science Foundation MoDL-SCALE project 2134168, the RPI-IBM Artificial Intelligence Research Collaboration (AIRC) and an Amazon Research Award.

References

Seyed Mehdi Alizadeh, Patrice Marcotte, and Gilles Savard. Two-stage stochastic bilevel programming over a transportation network. *Transportation Research Part B: Methodological*, 58:92–105, 2013.

João Carlos Alves Barata and Mahir Saleh Hussein. The Moore–Penrose pseudoinverse: A tutorial review of the theory. *Brazilian Journal of Physics*, 42(1):146–165, 2012.

Jerome Bracken and James T McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21(1):37–44, 1973.

Luce Brotcorne, Saïd Hanafi, and Raïd Mansi. One-level reformulation of the bilevel knapsack problem using dynamic programming. *Discrete Optimization*, 10(1):1–10, 2013.

Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.

Tianyi Chen, Yuejiao Sun, Quan Xiao, and Wotao Yin. A single-timescale method for stochastic bilevel optimization. In *Proc. International Conference on Artificial Intelligence and Statistics*, virtual, 2022a.

Xuxing Chen, Minhui Huang, and Shiqian Ma. Decentralized bilevel optimization. *arXiv preprint arXiv:2206.05670*, 2022b.

Ziyi Chen, Bhavya Kailkhura, and Yi Zhou. A fast and convergent proximal algorithm for regularized nonconvex and nonsmooth bi-level optimization. *arXiv preprint arXiv:2203.16615*, 2022c.

Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of operations research*, 153(1):235–256, 2007.

Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Stephan Dempe and Alain B Zemkoho. The bilevel programming problem: reformulations, constraint qualifications and optimality conditions. *Mathematical Programming*, 138(1):447–473, 2013.

Stephan Dempe, Joydeep Dutta, and Boris Mordukhovich. New necessary optimality conditions in optimistic bilevel programming. *Optimization*, 56(5-6):577–604, 2007.

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

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. International Conference on Machine Learning*, pages 1126–1135, Sydney, Australia, 2017.

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

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimilano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *Proc. International Conference on Machine Learning*, Stockholm, Sweden, 2018.

Hongchang Gao, Bin Gu, and My T Thai. Stochastic bilevel distributed optimization over a network. *arXiv preprint arXiv:2206.15025*, 2022a.

Lucy L Gao, Jane Ye, Haian Yin, Shangzhi Zeng, and Jin Zhang. Value function based difference-of-convex algorithm for bilevel hyperparameter selection problems. In *Proc. International Conference on Machine Learning*, pages 7164–7182, Baltimore, MD, 2022b.

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

Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1):267–305, 2016.

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

Stephen Gould, Dylan Campbell, Itzik Ben-Shabat, Chamin Hewa Koneputugodage, and Zhiwei Xu. Exploiting problem structure in deep declarative networks: Two case studies. *arXiv preprint arXiv:2202.12404*, 2022.

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

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

Yankun Huang, Qihang Lin, Nick Street, and Stephen Baek. Federated learning on adaptively weighted nodes by bilevel optimization. *arXiv preprint arXiv:2207.10751*, 2022.

M James. The generalised inverse. *The Mathematical Gazette*, 62(420):109–114, 1978.

Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *Proc. International Conference on Machine Learning*, virtual, 2021.

Kaiyi Ji, Mingrui Liu, Yingbin Liang, and Lei Ying. Will bilevel optimizers benefit from loops. *arXiv preprint arXiv:2205.14224*, 2022.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.

Ganesh Ramachandra Kini, Orestis Paraskevas, Samet Oymak, and Christos Thrampoulidis. Label-imbalanced and group-sensitive classification under overparameterization. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.

Junyi Li, Bin Gu, and Heng Huang. A fully single loop algorithm for bilevel optimization without hessian inverse. In *Proc. Association for the Advancement of Artificial Intelligence*, pages 7426–7434, virtual, 2022a.

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

Mingchen Li, Xuechen Zhang, Christos Thrampoulidis, Jiasi Chen, and Samet Oymak. Autobalance: Optimized loss functions for imbalanced data. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.

Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *Proc. International Conference on Learning Representations*, New Orleans, LA, 2019.

Risheng Liu, Jiaxin Gao, Jin Zhang, Deyu Meng, and Zhouchen Lin. Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021a.

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

Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. In *Proc. International Conference on Artificial Intelligence and Statistics*, pages 1540–1552, virtual, 2020.

Songtao Lu, Xiaodong Cui, Mark S Squillante, Brian Kingsbury, and Lior Horesh. Decentralized bilevel optimization for personalized client learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5543–5547, Singapore, 2022. IEEE.

Giulia Luise, Alessandro Rudi, Massimiliano Pontil, and Carlo Ciliberto. Differential properties of sinkhorn approximation for learning with wasserstein distance. *Proc. Advances in Neural Information Processing Systems*, 31, 2018.

Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *Proc. International Conference on Machine Learning*, pages 2113–2122, Lille, France, 2015.

Patrice Marcotte. Network design problem with congestion effects: A case of bilevel programming. *Mathematical programming*, 34(2):142–162, 1986.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proc. International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, Fort Lauderdale, FL, 2017.

Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtárik. Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally! In *Proc. International Conference on Machine Learning*, Baltimore, MD, 2022.

Aritra Mitra, Rayana Jaafar, George J Pappas, and Hamed Hassani. Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients. *Advances in Neural Information Processing Systems*, 34:14606–14619, 2021.

Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014.

Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: from classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2017.

Sebastian U Stich. Local SGD converges fast and communicates little. In *Proc. International Conference on Learning Representations*, New Orleans, LA, 2019.

Sebastian U Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for SGD with delayed gradients and compressed updates. *Journal of Machine Learning Research*, 21:1–36, 2020.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Davoud Ataee Tarzanagh, Mingchen Li, Christos Thrampoulidis, and Samet Oymak. FEDNEST: Federated bilevel, minimax, and compositional optimization. In *Proc. International Conference on Machine Learning*, Baltimore, MD, 2022.

Ioannis Tsaknakis, Prashant Khanduri, and Mingyi Hong. An implicit gradient-type method for linearly constrained bilevel problems. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5438–5442, Singapore, 2022.

Luis N Vicente and Paul H Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global optimization*, 5(3):291–306, 1994.

Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local SGD better than minibatch SGD? In *Proc. International Conference on Machine Learning*, pages 10334–10343, virtual, 2020.

Haibo Yang, Minghong Fang, and Jia Liu. Achieving linear speedup with partial worker participation in non-iid federated learning. In *Proc. International Conference on Learning Representations*, virtual, 2021a.

Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. In *Proc. Advances in Neural Information Processing Systems*, pages 13670–13682, virtual, 2021b.

Shuoguang Yang, Xuezhou Zhang, and Mengdi Wang. Decentralized gossip-based stochastic bilevel optimization over communication networks. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2022.

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

Jane J Ye and Daoli Zhu. New necessary optimality conditions for bilevel programs by combining the MPEC and value function approaches. *SIAM Journal on Optimization*, 20(4):1885–1905, 2010.

Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proc. Association for the Advancement of Artificial Intelligence*, pages 5693–5700, Honolulu, HI, 2019.

Supplementary Material for "Alternating Projected SGD for Equality-constrained Bilevel Optimization"

Table of Contents

A Preliminaries

Definition 10. *Suppose $\mathcal{L}:\mathbb{R}^{d_1}\to\mathbb{R}^{d_2}$ such that each of its first-order partial derivatives exist on $\mathbb{R}^d$, then its Jacobian is defined as*

$$\nabla\mathcal{L}=\begin{bmatrix}\frac{\partial\mathcal{L}_1}{\partial x_1} & \cdots & \frac{\partial\mathcal{L}_1}{\partial x_{d_1}}\\ \vdots & \ddots & \vdots\\ \frac{\partial\mathcal{L}_{d_2}}{\partial x_1} & \cdots & \frac{\partial\mathcal{L}_{d_2}}{\partial x_{d_1}}\end{bmatrix}. \tag{22}$$

Therefore, $\nabla h(x)$ and $\nabla y^*(x)$ can be written as

$$\nabla h(x)=\begin{bmatrix}\frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial\mathcal{L}_1}{\partial x_{d_x}}\\ \vdots & \ddots & \vdots\\ \frac{\partial\mathcal{L}_{m_y}}{\partial x_1} & \cdots & \frac{\partial\mathcal{L}_{m_y}}{\partial x_{d_x}}\end{bmatrix},\quad \nabla y^*(x)=\begin{bmatrix}\frac{\partial y_1^*(x)}{\partial x_1} & \cdots & \frac{\partial y_1^*(x)}{\partial x_{d_x}}\\ \vdots & \ddots & \vdots\\ \frac{\partial y_{d_y}^*(x)}{\partial x_1} & \cdots & \frac{\partial y_{d_y}^*(x)}{\partial x_{d_x}}\end{bmatrix}. \tag{23}$$

Definition 11. *The Bregman divergence of a differentiable function $\mathcal{L}:\mathbb{R}^d\to\mathbb{R}$ is defined as*

$$D_{\mathcal{L}}(u,v):=\mathcal{L}(u)-\mathcal{L}(v)-\langle\nabla\mathcal{L}(v),u-v\rangle.$$

For an L-smooth and μ-strongly convex function $\mathcal{L}$, we have

$$\frac{\mu}{2}\|u-v\|^2 \leq D_{\mathcal{L}}(u,v) \leq \frac{L}{2}\|u-v\|^2 \tag{24}$$

and

$$\frac{1}{2L}\|\nabla\mathcal{L}(u)-\nabla\mathcal{L}(v)\|^2 \leq D_{\mathcal{L}}(u,v) \leq \frac{1}{2\mu}\|\nabla\mathcal{L}(u)-\nabla\mathcal{L}(v)\|^2. \tag{25}$$

Moreover, for any u, v, we have $\langle\nabla\mathcal{L}(u)-\nabla\mathcal{L}(v), u-v\rangle = D_{\mathcal{L}}(u,v)+D_{\mathcal{L}}(v,u)$.

Lemma 12 (Closed form linear operator of projection). *For any nonempty linear space $C=\{z \mid Az+b=0\}$, the projection operator has the following closed form*

$$\mathrm{Proj}_C(x) = (I-A^\dagger A)x - A^\dagger b \tag{26}$$

where $A^\dagger$ is the Moore-Penrose inverse of A.

Proof. **Case 1.** We first consider the case where $b=0$.

We denote $P=I-A^\dagger A$, then $C=\mathrm{Ker}(A)$. According to Proposition 3.3. in (Barata and Hussein, 2012), we know that P is an orthogonal projection and

$$C=\mathrm{Ker}(A)=\mathrm{Ran}(P),\ \ C^\perp=\mathrm{Ker}(A)^\perp=\mathrm{Ran}(A^\dagger)\overset{(a)}{=}\mathrm{Ran}(A^\dagger A)=\mathrm{Ran}(I-P)$$

where (a) holds since $A^\dagger = A^\dagger A A^\dagger$ (Barata and Hussein, 2012) and

$$\forall z = A^\dagger A w, z = A^\dagger(Aw) \Rightarrow \mathrm{Ran}(A^\dagger A)\subset\mathrm{Ran}(A^\dagger),$$
$$\forall z = A^\dagger w, z = A^\dagger A(A^\dagger w) \Rightarrow \mathrm{Ran}(A^\dagger)\subset\mathrm{Ran}(A^\dagger A).$$

Thus, for any x, we can write $x=Px+(I-P)x$ and $Px\perp(I-P)x$, which means Px is the orthogonal projection. Then due to the uniqueness of the orthogonal decomposition, $\mathrm{Proj}_C(x)=Px$.

Case 2. We then consider the case when $b\neq 0$.

For any $z\in C$, we have $z+A^\dagger b\in\mathrm{Ker}(A)$ since

$$A(z+A^\dagger b)=Az+AA^\dagger b=-b+AA^\dagger b\overset{(a)}{=}0$$

where (a) holds since C is nonempty if and only if $AA^\dagger b=b$ (James, 1978). Similarly, we can prove for any $z\in\mathrm{Ker}(A)$, $z-A^\dagger b\in C$. Thus, $\mathrm{Ker}(A)=C+A^\dagger b$.

Moreover, since projection operator minimizes the distance to set C, we have

$$\begin{aligned}
\mathrm{Proj}_C(x) &= \arg\min_{z\in C}\|z-x\|^2 = \arg\min_{z\in C}\|z+A^\dagger b-(x+A^\dagger b)\|^2\\
&\overset{(a)}{=}\left\{\arg\min_{w\in\mathrm{Ker}(A)}\|w-(x+A^\dagger b)\|^2\right\}-A^\dagger b\\
&=\mathrm{Proj}_{\mathrm{Ker}(A)}(x+A^\dagger b)-A^\dagger b\\
&\overset{(b)}{=}P(x+A^\dagger b)-A^\dagger b\\
&\overset{(c)}{=}(I-A^\dagger A)x-A^\dagger b
\end{aligned}$$

where (a) is due to $\mathrm{Ker}(A)=C+A^\dagger b$, (b) comes from Case 1, and (c) is derived from the definition of P and $(I-A^\dagger A)A^\dagger b=(A^\dagger-A^\dagger AA^\dagger)b=0$. □

B Proof for Lemma 1

Lemma 13 (Stationarity measure under linear equality constraints). *For any $x \in \mathcal{X}$ and any $\lambda > 0$, it holds that*

$$\|\nabla F(x)\|_{P_x}^2 = \|\lambda^{-1}(x - \mathrm{Proj}_{\mathcal{X}}(x - \lambda \nabla F(x)))\|^2$$

where $P_x = I - B^\dagger B$.

Proof. For any λ, according to (26), we have

$$\begin{aligned}
\lambda^{-1}(x - \mathrm{Proj}_{\mathcal{X}}(x - \lambda \nabla F(x))) &= \lambda^{-1}(x - (I - B^\dagger B)(x - \lambda \nabla F(x)) - B^\dagger e) \\
&= \lambda^{-1}(x + (I - B^\dagger B)\lambda \nabla F(x) - [(I - B^\dagger B)x + B^\dagger e]) \\
&= \lambda^{-1}(x - \mathrm{Proj}_{\mathcal{X}}(x)) + (I - B^\dagger B)\nabla F(x) \\
&= (I - B^\dagger B)\nabla F(x)
\end{aligned}$$

where the first equality is due to $\mathcal{X} = \{x | Bx = e\}$ and (26) the last two equality holds since $x \in \mathcal{X}$ and (26). Then with $(I - B^\dagger B)^2 = I - B^\dagger B$ and the definition of $\|\cdot\|_{P_x}$, the proof is complete. □

C Proof for Algorithm 1

In this section, we present the proof of Algorithm 1.

In this section, we define

$$\mathcal{F}_k^s := \sigma\{y^0, x^0, \cdots, y^k, x^k, y^{k,1}, \cdots, y^{k,s}\} \tag{27}$$

where $\sigma\{\cdot\}$ denotes the σ-algebra generated by the random variables. Then it follows that $\mathcal{F}_k^S = \sigma\{y^0, x^0, \cdots, y^{k+1}\}$.

C.1 Proof of Lemma 2 and Lemma 3

We restate Lemma 2 and Lemma 3 together in a more formal way.

Restatement of Lemma 2-3. *Under Assumption 1–2 and 4, the gradient of $y^*(x)$ can be expressed as*

$$\nabla y^*(x) = -V_2(V_2^\top \nabla_{yy} g(x, y^*(x)) V_2)^{-1} V_2^\top \left(\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\right) - A^\dagger \nabla h(x) \tag{28}$$

where V_2 is the orthogonal basis of $\mathrm{Ker}(A) := \{y \mid Ay = 0\}$. Moreover, $y^*(x)$ is L_y Lipschitz continuous and L_{yx} smooth with

$$\begin{aligned}
L_y &:= \frac{\ell_{g,1} + (\ell_{g,1} + \mu_g)\|A^\dagger\|\ell_{h,0}}{\mu_g}, \\
L_{yx} &:= \frac{\ell_{g,2}\left(1 + \|A^\dagger\|\ell_{h,0}\right)(1 + L_y)(1 + \frac{\ell_{g,1}}{\mu_g}) + (\ell_{g,1} + \mu_g)\|A^\dagger\|\ell_{h,1}}{\mu_g}.
\end{aligned} \tag{29}$$

Proof. By singular value decomposition, we can decompose $A = U\Sigma V^\top$ with $\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{m_y \times d_y}$, and orthogonal matrix $U = [U_1\ U_2] \in \mathbb{R}^{m_y \times m_y}$ and $V = [V_1\ V_2] \in \mathbb{R}^{d_y \times d_y}$. Also, by assuming $\mathrm{Rank}(A) = r$, we know that $U_1 \in \mathbb{R}^{m_y \times r}, V_1 \in \mathbb{R}^{d_y \times r}$ and $\Sigma_1 \in \mathbb{R}^{r \times r}$ are full rank submatrix. Therefore, A can be decomposed by

$$A = [U_1\ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^\top \\ V_2^\top \end{bmatrix} = [U_1\Sigma_1 \quad 0] \begin{bmatrix} V_1^\top \\ V_2^\top \end{bmatrix} = U_1\Sigma_1 V_1^\top$$

and V_2 is the orthogonal basis of $\mathrm{Ker}(A)$.

Next, if we define $y_0(x) := A^\dagger(c - h(x))$, we can prove $y_0(x) \in \mathcal{Y}(x)$ since

$$Ay_0(x) = AA^\dagger(c - h(x)) = c - h(x)$$

where the last equality holds due to Assumption 4. From the definition of $y_0(x)$ and (23), we know

$$\nabla_x y_0(x) = -A^\dagger \nabla h(x). \tag{30}$$

Moreover, since V_2 is the orthogonal basis of $\mathrm{Ker}(A)$, we know $\mathcal{Y}(x) = y_0(x) + \mathrm{Ran}(V_2)$. Thus, let $z^*(x) = \arg\min_z g(x, y_0(x) + V_2 z)$, then we have $y^*(x) = y_0(x) + V_2 z^*(x)$.

Since $z^*(x)$ satisfies

$$\nabla_z g(x, y_0(x) + V_2 z^*(x)) = V_2^\top \nabla_y g(x, y_0(x) + V_2 z^*(x)) = 0$$

then taking the gradient with respect to x of both sides, we get

$$\begin{aligned}
0 &= \nabla_x (V_2^\top \nabla_y g(x, y_0(x) + V_2 z^*(x))) \\
&= \nabla_{xy} g(x, y_0(x) + V_2 z^*(x)) V_2 + \left(\nabla_x z^*(x)^\top V_2^\top + \nabla_x y_0(x)^\top\right) \nabla_{yy} g(x, y_0(x) + V_2 z^*(x)) V_2 \\
&= \nabla_{xy} g(x, y_0(x) + V_2 z^*(x)) V_2 + \left(\nabla_x z^*(x)^\top V_2^\top - \nabla^\top h(x) A^{\dagger\top}\right) \nabla_{yy} g(x, y_0(x) + V_2 z^*(x)) V_2 \\
&= \nabla_{xy} g(x, y^*(x)) V_2 + \left(\nabla_x z^*(x)^\top V_2^\top - \nabla^\top h(x) A^{\dagger\top}\right) \nabla_{yy} g(x, y^*(x)) V_2
\end{aligned} \tag{31}$$

where the third equality holds from (30). Then, rearranging (31), we get

$$\nabla z^*(x) = -\left(V_2^\top \nabla_{yy} g(x, y^*(x)) V_2\right)^{-1} V_2^\top \left(\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\right) \tag{32}$$

and as a result of $y^*(x) = y_0(x) + V_2 z^*(x)$, we have

$$\begin{aligned}
\nabla y^*(x) &= \nabla y_0(x) + V_2 \nabla z^*(x) \\
&= -A^\dagger \nabla h(x) - V_2 \left(V_2^\top \nabla_{yy} g(x, y^*(x)) V_2\right)^{-1} V_2^\top \left(\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\right).
\end{aligned} \tag{33}$$

Next, utilizing the fact that V_2 is the orthogonal matrix, we know $\mu_g I_{d_y - r} \preceq V_2^\top \nabla_{yy} g(x, y) V_2$. Therefore, we have for any x, y,

$$V_2 \left(V_2^\top \nabla_{yy} g(x, y) V_2\right)^{-1} V_2^\top \preceq \frac{1}{\mu_g} I. \tag{34}$$

Besides, we have for any x, it follows that

$$\begin{aligned}
\|\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\| &\le (1 + \|A^\dagger\| \ell_{h,0}) \|\nabla^2 g(x, y^*(x))\| \\
&\le (1 + \|A^\dagger\| \ell_{h,0}) \ell_{g,1}.
\end{aligned} \tag{35}$$

As a result of (33), (34) and (35), $\nabla y^*(x)$ is bounded by

$$\begin{aligned}
&\|\nabla y^*(x)\| \\
&\le \left\| V_2 \left(V_2^\top \nabla_{yy} g(x, y^*(x)) V_2\right)^{-1} V_2^\top \right\| \|\nabla_{yx} g(x, y^*(x)) - \nabla_{yy} g(x, y^*(x)) A^\dagger \nabla h(x)\| + \|A^\dagger \nabla h(x)\| \\
&\le \frac{\ell_{g,1} + (\ell_{g,1} + \mu_g)\|A^\dagger\| \ell_{h,0}}{\mu_g} = L_y
\end{aligned}$$

which implies $y^*(x)$ is L_y Lipschitz continuous.

Finally, we aim to prove the smoothness of $y^*(x)$. Defining $B_1 = V_2^\top \nabla_{yy} g(x_1, y^*(x_1)) V_2$ and $B_2 = V_2^\top \nabla_{yy} g(x_2, y^*(x_2)) V_2$, for any x_1 and x_2, we have

$$\begin{aligned}
&\|\nabla y^*(x_1) - \nabla y^*(x_2)\| \\
&= \Big\| V_2 \left(V_2^\top \nabla_{yy} g(x_1, y^*(x_1)) V_2\right)^{-1} V_2^\top \left(\nabla_{yx} g(x_1, y^*(x_1)) - \nabla_{yy} g(x_1, y^*(x_1)) A^\dagger \nabla h(x_1)\right) \\
&\quad - V_2 \left(V_2^\top \nabla_{yy} g(x_2, y^*(x_2)) V_2\right)^{-1} V_2^\top \left(\nabla_{yx} g(x_2, y^*(x_2)) - \nabla_{yy} g(x_2, y^*(x_2)) A^\dagger \nabla h(x_2)\right) \Big\| \\
&\quad + \|A^\dagger (\nabla h(x_1) - \nabla h(x_2))\|
\end{aligned}$$

$$
\begin{aligned}
&\leq \|V_2 B_1^{-1} V_2^\top\| \|\nabla_{yx} g(x_1, y^*(x_1)) - \nabla_{yx} g(x_2, y^*(x_2)))\| \\
&\quad + \|V_2 B_1^{-1} V_2^\top\| \|\nabla_{yy} g(x_1, y^*(x_1)) A^\dagger \nabla h(x_1) - \nabla_{yy} g(x_2, y^*(x_2)) A^\dagger \nabla h(x_2)\| \\
&\quad + \|V_2 (B_1^{-1} - B_2^{-1}) V_2^\top\| \|\nabla_{yx} g(x_2, y^*(x_2)) - \nabla_{yy} g(x_2, y^*(x_2)) A^\dagger \nabla h(x_2)\| + \ell_{h,1} \|A^\dagger\| \|x_1 - x_2\| \\
&\overset{(a)}{\leq} \frac{1}{\mu_g} \|\nabla_{yx} g(x_1, y^*(x_1)) - \nabla_{yx} g(x_2, y^*(x_2))\| \\
&\quad + \frac{1}{\mu_g} \|\nabla_{yy} g(x_1, y^*(x_1)) - \nabla_{yy} g(x_2, y^*(x_2))\| \|A^\dagger\| \|\nabla h(x_1)\| \\
&\quad + \frac{1}{\mu_g} \|\nabla_{yy} g(x_2, y^*(x_2))\| \|A^\dagger\| \|\nabla h(x_1) - \nabla h(x_2)\| \\
&\quad + \frac{\ell_{g,1} \left(1 + \|A^\dagger\| \ell_{h,0}\right)}{\mu_g^2} \|\nabla_{yy} g(x_1, y^*(x_1)) - \nabla_{yy} g(x_2, y^*(x_2))\| + \ell_{h,1} \|A^\dagger\| \|x_1 - x_2\| \\
&\overset{(b)}{\leq} \frac{\ell_{g,2} \left(1 + \|A^\dagger\| \ell_{h,0}\right) (1 + L_y)(1 + \frac{\ell_{g,1}}{\mu_g}) + (\ell_{g,1} + \mu_g) \|A^\dagger\| \ell_{h,1}}{\mu_g} \|x_1 - x_2\|
\end{aligned} \tag{36}
$$

where (a) comes from (34), (35) and the following fact

$$
\begin{aligned}
&V_2 \left(B_1^{-1} - B_2^{-1}\right) V_2^\top \\
&= V_2 B_1^{-1} \left(B_2 - B_1\right) B_2^{-1} V_2^\top \\
&= V_2 B_1^{-1} \left(\left(V_2^\top \nabla_{yy} g(x_2, y^*(x_2)) V_2\right) - \left(V_2^\top \nabla_{yy} g(x_1, y^*(x_1)) V_2\right)\right) B_2^{-1} V_2^\top \\
&= V_2 B_1^{-1} V_2^\top \left(\nabla_{yy} g(x_2, y^*(x_2)) - \nabla_{yy} g(x_1, y^*(x_1))\right) V_2 B_2^{-1} V_2^\top
\end{aligned}
$$

so that

$$
\begin{aligned}
\|V_2 \left(B_1^{-1} - B_2^{-1}\right) V_2^\top\| &\leq \|V_2 B_1^{-1} V_2^\top\| \|\nabla_{yy} g(x_2, y^*(x_2)) - \nabla_{yy} g(x_1, y^*(x_1))\| \|V_2 B_2^{-1} V_2^\top\| \\
&\leq \frac{1}{\mu_g^2} \|\nabla_{yy} g(x_2, y^*(x_2)) - \nabla_{yy} g(x_1, y^*(x_1))\|
\end{aligned} \tag{37}
$$

and (b) comes from

$$
\begin{aligned}
\|\nabla^2 g(x_1, y^*(x_1)) - \nabla^2 g(x_2, y^*(x_2))\| &\leq \ell_{g,2} \left[\|x_1 - x_2\| + \|y^*(x_1) - y^*(x_2)\|\right] \\
&\leq \ell_{g,2} \left(1 + L_y\right) \|x_1 - x_2\|
\end{aligned} \tag{38}
$$

from which the proof is complete. □

C.2 Smoothness of $F(x)$

Lemma 14. *Under Assumption 1–4, $F(x)$ is smooth with constant L_F which is defined as*

$$
L_F := \ell_{f,1} \left(1 + L_y\right)^2 + \ell_{f,0} L_{yx} \tag{39}
$$

Proof. For any x_1 and x_2, we have that

$$
\begin{aligned}
\|\nabla F(x_1) - \nabla F(x_2)\| &= \|\nabla_x f(x_1, y^*(x_1)) + \nabla^\top y^*(x_1) \nabla_y f(x_1, y^*(x_1)) \\
&\quad - \nabla_x f(x_2, y^*(x_2)) + \nabla^\top y^*(x_2) \nabla_y f(x_2, y^*(x_2))\| \\
&\leq \|\nabla_x f(x_1, y^*(x_1)) - \nabla_x f(x_2, y^*(x_2))\| \\
&\quad + \|\nabla^\top y^*(x_1) \nabla_y f(x_1, y^*(x_1)) - \nabla^\top y^*(x_2) \nabla_y f(x_2, y^*(x_2))\| \\
&\leq \ell_{f,1} \left(\|x_1 - x_2\| + \|y^*(x_1) - y^*(x_2)\|\right) \\
&\quad + \|\nabla y^*(x_1)\| \|\nabla_y f(x_1, y^*(x_1)) - \nabla_y f(x_2, y^*(x_2))\| \\
&\quad + \|\nabla_y f(x_2, y^*(x_2))\| \|\nabla y^*(x_1) - \nabla y^*(x_2)\| \\
&\overset{(a)}{\leq} \left(\ell_{f,1} \left(1 + L_y\right)^2 + \ell_{f,0} L_{yx}\right) \|x_1 - x_2\| = L_F \|x_1 - x_2\|
\end{aligned}
$$

where (a) comes frome the Lipschitz continuity of $y^*(x), \nabla y^*(x)$ in Lemma 3 and the Lipschitz continuity of ∇f and f in Assumption 1. □

C.3 Supportive lemmas for the proof of Theorem 4

For simplicity, we denote

$$\begin{aligned}\overline{\nabla} f(x,y) = \nabla_x f(x,y) + \big[\big(\nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y) - \nabla_{xy} g(x,y)\big) \\ \times V_2 (V_2^\top \nabla_{yy} g(x,y) V_2)^{-1} V_2^\top - \nabla h(x)^\top A^{\dagger\top}\big] \nabla_y f(x,y).\end{aligned} \tag{40}$$

Lemma 15 (Boundness of $\overline{\nabla} f(x,y)$). *For any x, y, we have that $\|\overline{\nabla} f(x,y)\| \le \ell_{f,0}(1+L_y)$.*

Proof. Based on (34), we have that

$$\left\| V_2 \left(V_2^\top \nabla_{yy} g(x,y) V_2\right)^{-1} V_2^\top \right\| \le \frac{1}{\mu_g}. \tag{41}$$

Then we can obtain the bound for $\overline{\nabla} f(x,y)$ since

$$\begin{aligned}\|\overline{\nabla} f(x,y)\| &\le \|\nabla_x f(x,y)\| + \|\nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y) - \nabla_{xy} g(x,y)\| \\ &\quad \times \|V_2 (V_2^\top \nabla_{yy} g(x,y) V_2)^{-1} V_2^\top\| \|\nabla_y f(x,y)\| + \|\nabla h(x)^\top A^{\dagger\top}\| \|\nabla_y f(x,y)\| \\ &\le \ell_{f,0} \left(1 + \frac{\ell_{g,1}\left(1 + \ell_{h,0}\|A^\dagger\|\right)}{\mu_g} + \ell_{h,0}\|A^\dagger\|\right) = \ell_{f,0}(1+L_y)\end{aligned}$$

from which the proof is complete. □

Lemma 16 (Lipschitz continuity of $\overline{\nabla} f(x,y)$). *Under Assumption 1–2 and 4, $\overline{\nabla} f(x,y)$ is L_f Lipschitz continuous with respect to y, where the constant is defined as*

$$L_f := \left(1 + \ell_{h,0}\|A^\dagger\|\right) \left(\ell_{f,1} + \frac{\ell_{g,1}\ell_{f,1} + \ell_{f,0}\ell_{g,2}}{\mu_g} + \frac{\ell_{f,0}\ell_{g,1}\ell_{g,2}}{\mu_{g,1}^2}\right). \tag{42}$$

Proof. For simplicity, we define some notations first.

$$\begin{aligned}&B_1 = V_2^\top \nabla_{yy} g(x,y_1) V_2, \quad B_2 = V_2^\top \nabla_{yy} g(x,y_2) V_2, \\ &C_1 = \nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y_1) - \nabla_{xy} g(x,y_1), \\ &C_2 = \nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y_2) - \nabla_{xy} g(x,y_2).\end{aligned}$$

For $i = 1, 2$, according to (34), we have the following bounds.

$$\|C_i\| \le \left(1 + \ell_{h,0}\|A^\dagger\|\right) \ell_{g,1}, \quad \|V_2 B_i^{-1} V_2^\top\| \le \frac{1}{\mu_g}, \quad \|\nabla_y f(x,y_i)\| \le \ell_{f,0}. \tag{43}$$

Besides, we can also bound their differences as

$$\|C_1 - C_2\| \le \left(1 + \|A^\dagger\|\ell_{h,0}\right) \|\nabla^2 g(x,y_1) - \nabla^2 g(x,y_2)\| \le \left(1 + \|A^\dagger\|\ell_{h,0}\right) \ell_{g,2}\|x_1 - x_2\|$$

and

$$\begin{aligned}\|V_2 B_1^{-1} V_2^\top - V_2 B_2^{-1} V_2^\top\| &\overset{(a)}{\le} \|V_2 B_1^{-1} V_2^\top\| \|V_2 B_2^{-1} V_2^\top\| \|\nabla_{yy} g(x,y_1) - \nabla_{yy} g(x,y_2)\| \\ &\le \frac{\ell_{g,2}}{\mu_g^2}\|y_1 - y_2\|\end{aligned}$$

where (a) is due to

$$\begin{aligned}&V_2 \left(B_1^{-1} - B_2^{-1}\right) V_2^\top \\ &= V_2 B_1^{-1} \left(B_2 - B_1\right) B_2^{-1} V_2^\top \\ &= V_2 B_1^{-1} \left(\left(V_2^\top \nabla_{yy} g(x,y_2) V_2\right) - \left(V_2^\top \nabla_{yy} g(x,y_1) V_2\right)\right) B_2^{-1} V_2^\top\end{aligned}$$

$$= V_2 B_1^{-1} V_2^\top \left(\nabla_{yy} g(x, y_2) - \nabla_{yy} g(x, y_1)\right) V_2 B_2^{-1} V_2^\top. \tag{44}$$

Likewise, we have

$$\|\nabla_y f(x, y_1) - \nabla_y f(x, y_2)\| \leq \ell_{f,1} \|x_1 - x_2\|. \tag{45}$$

Thus, for any x, y_1, y_2, based on (43) and (45), we have

$$\begin{aligned}
&\|\overline{\nabla} f(x, y_1) - \overline{\nabla} f(x, y_2)\| \\
\leq& \|\nabla_x f(x, y_1) - \nabla_x f(x, y_2)\| + \|C_1 V_2 B_1^{-1} V_2^\top \nabla_y f(x, y_1) - C_2 V_2 B_2^{-1} V_2^\top \nabla_y f(x, y_2)\| \\
&+ \|\nabla h(x)\| \|A^\dagger\| \|\nabla_y f(x, y_1) - \nabla_y f(x, y_2)\| \\
\overset{(a)}{\leq}& (1 + \ell_{h,0} \|A^\dagger\|) \ell_{f,1} \|y_1 - y_2\| + \|C_1\| \|V_2 B_1^{-1} V_2^\top\| \|\nabla_x f(x, y_1) - \nabla_x f(x, y_2)\| \\
&+ \|C_1\| \|\nabla_x f(x, y_2)\| \|V_2 B_1^{-1} V_2^\top - V_2 B_2^{-1} V_2^\top\| + \|V_2 B_2^{-1} V_2^\top\| \|\nabla_x f(x, y_2)\| \|C_1 - C_2\| \\
\overset{(b)}{\leq}& \left(1 + \ell_{h,0} \|A^\dagger\|\right) \left(\ell_{f,1} + \frac{\ell_{g,1} \ell_{f,1} + \ell_{f,0} \ell_{g,2}}{\mu_g} + \frac{\ell_{f,0} \ell_{g,1} \ell_{g,2}}{\mu_{g,1}^2}\right) \|y_1 - y_2\|
\end{aligned} \tag{46}$$

where (a) is due to

$$\begin{aligned}
&C_1 D_1 E_1 - C_2 D_2 E_2 \\
=&C_1 D_1 E_1 - C_1 D_1 E_2 + C_1 D_1 E_2 - C_1 D_2 E_2 + C_1 D_2 E_2 - C_2 D_2 E_2 \\
=&C_1 D_1 (E_1 - E_2) + C_1 E_2 (D_1 - D_2) + D_2 E_2 (C_1 - C_2)
\end{aligned} \tag{47}$$

(b) comes from (43) and (45), from which the proof is complete. □

To prove the bias and variance of gradient estimator h_f^k, we leverage the following fact.

Lemma 17 ((Hong et al., 2020, Lemma 12)). *Let Z_i be a sequence of stochastic matrix defined recursively as $Z_i = Y_i Z_{i-1}, i \geq 0$ with $Z_{-1} = I \in \mathbb{R}^{d \times d}$, Y_i are independent, symmetric random matrix satisfying that*

$$\|\mathbb{E}[Y_i]\| \leq 1 - \mu, \ \ \mathbb{E}\left[\|Y_i - \mathbb{E}[Y_i]\|^2\right] \leq \sigma^2.$$

If $(1-\mu)^2 + \sigma^2 < 1$, then for any $i > 0$, it holds that

$$\mathbb{E}\left[\|Z_i\|^2\right] \leq d\left((1-\mu)^2 + \sigma^2\right)^i.$$

Based on this lemma, we can bound the second moment bound of Hessian inverse estimator.

Lemma 18. *Let $\tilde{c} = \frac{\mu_g}{\mu_g^2 + \sigma_{g,2}^2}$ and for any k, denote the Hessian inverse estimator as*

$$H_{yy}^k = \frac{\tilde{c} N}{\ell_{g,1}} \prod_{n=0}^{N'} \left(I - \frac{\tilde{c}}{\ell_{g,1}} V_2^\top \nabla_{yy}^2 g\left(x^k, y^{k+1}; \phi_{(n)}^k\right) V_2\right).$$

Then the second moment bound of H_{yy}^k can be bounded as

$$\mathbb{E}\left[\|H_{yy}^k\|^2 | \mathcal{F}_k^S\right] \leq \frac{N(d_y - r)}{\ell_{g,1}(\mu_g^2 + \sigma_{g,2}^2)}.$$

Proof. Let $Y_n = I - \frac{\tilde{c}}{\ell_{g,1}} V_2^\top \nabla_{yy}^2 g\left(x, y; \phi_{(n)}\right) V_2$, we know that

$$\|\mathbb{E}[Y_n]\| \leq \left(1 - \frac{\tilde{c}\mu_g}{\ell_{g,1}}\right), \ \ \mathbb{E}\left[\|Y_n - \mathbb{E}[Y_n]\|^2\right] \leq \frac{\tilde{c}^2 \sigma_{g,2}^2}{\ell_{g,1}^2}$$

Moreover, since

$$\left(1 - \frac{\tilde{c}\mu_g}{\ell_{g,1}}\right)^2 + \frac{\tilde{c}^2 \sigma_{g,2}^2}{\ell_{g,1}^2} = 1 - \frac{2\tilde{c}\mu_g}{\ell_{g,1}} + \frac{\tilde{c}^2(\mu_g^2 + \sigma_{g,2}^2)}{\ell_{g,1}^2} = 1 - \frac{\mu_g^2}{\ell_{g,1}\left(\mu_g^2 + \sigma_{g,2}^2\right)} < 1$$

which satisfies the condition in Lemma 17, we can then plugging Y_n into Lemma 17 and achieves the second moment bound for H_{yy}^k.

$$\begin{aligned}\mathbb{E}\left[\|H_{yy}^k\|^2|\mathcal{F}_k^S\right] &= \mathbb{E}\left[\mathbb{E}\left[\|H_{yy}^k\|^2|N',\mathcal{F}_k^S\right]|\mathcal{F}_k^S\right] \leq \mathbb{E}\left[\frac{\tilde{c}^2N^2(d_y-r)}{\ell_{g,1}^2}\left(1-\frac{\mu_g^2}{\ell_{g,1}\left(\mu_g^2+\sigma_{g,2}^2\right)}\right)^{N'}\Big|N'\right]\\ &\leq \frac{\tilde{c}^2N(d_y-r)}{\ell_{g,1}^2}\sum_{n=0}^{N-1}\left(1-\frac{\mu_g^2}{\ell_{g,1}\left(\mu_g^2+\sigma_{g,2}^2\right)}\right)^n\\ &\leq \frac{\tilde{c}^2N(d_y-r)}{\ell_{g,1}^2}\frac{\ell_{g,1}\left(\mu_g^2+\sigma_{g,2}^2\right)}{\mu_g^2} \overset{(a)}{\leq} \frac{N(d_y-r)}{\ell_{g,1}\left(\mu_g^2+\sigma_{g,2}^2\right)}.\end{aligned}$$

where r is rank of A and (a) comes from the choice of $\tilde{c}$. □

Lemma 19 (Bias and variance of gradient estimator). *Let* $\tilde{c}=\frac{\mu_g}{\mu_g^2+\sigma_{g,2}^2}$ *and define*

$$\bar{h}_f^k=\mathbb{E}\left[h_f^k|\,\mathcal{F}_k^S\right],$$

then h_f^k *is a biased estimator of UL gradient which satisfies that*

$$\|\bar{h}_f^k-\overline{\nabla}f(x^k,y^{k+1})\|\leq L_y\ell_{f,0}\left(1-\frac{\mu_g^2}{\ell_{g,1}(\mu_g^2+\sigma_{g,2}^2)}\right)^N=:b_k \tag{48}$$

$$\mathbb{E}\left[\|h_f^k-\bar{h}_f^k\|^2|\mathcal{F}_k^S\right]\leq\sigma_f^2+\frac{4N(1+\ell_{h,0}^2\|A^\dagger\|^2)(d_y-r)(\ell_{g,1}^2+\sigma_{g,2}^2)\left(2\sigma_f^2+\ell_{f,0}^2\right)}{\ell_{g,1}(\mu_g^2+\sigma_{g,2}^2)}=:\tilde{\sigma}_f^2=\mathcal{O}\left(N\kappa^2\right). \tag{49}$$

Proof. We first prove (48) by noticing that the error by finite updates can be bounded by

$$\left\|V_2\left((I-D)^{-1}-\sum_{n=0}^{N-1}D^n\right)V_2^\top\right\|=\left\|V_2\left(\sum_{n=N}^{\infty}D^n\right)V_2^\top\right\|=\left\|\sum_{n=N}^{\infty}(V_2DV_2^\top)^n\right\| \tag{50}$$

$$\leq\sum_{n=N}^{\infty}\|V_2DV_2^\top\|^n=\frac{\|V_2DV_2^\top\|^N}{1-\|V_2DV_2^\top\|} \tag{51}$$

Thus, letting $D=I-\frac{\tilde{c}}{\ell_{g,1}}V_2^\top\nabla_{yy}g(x,y)V_2$ and multiplying each side by $\frac{\tilde{c}}{\ell_{g,1}}$, we obtain that

$$\begin{aligned}&\left\|V_2\left(\left(V_2^\top\nabla_{yy}g(x,y)V_2\right)^{-1}-\frac{\tilde{c}}{\ell_{g,1}}\sum_{n=0}^{N-1}(I-\frac{\tilde{c}}{\ell_{g,1}}V_2^\top\nabla_{yy}g(x,y)V_2)^n\right)V_2^\top\right\|\\ \leq&\frac{\tilde{c}\|V_2\left(I-\frac{\tilde{c}}{\ell_{g,1}}\nabla_{yy}g(x,y)\right)V_2^\top\|^N}{\ell_{g,1}\left(1-\|V_2\left(I-\frac{\tilde{c}}{\ell_{g,1}}\nabla_{yy}g(x,y)\right)V_2^\top\|\right)}\\ \leq&\frac{\left(1-\frac{\tilde{c}\mu_g}{\ell_{g,1}}\right)^N}{\mu_g}\end{aligned} \tag{52}$$

where the second inequality holds according to $\mu_gI_{d_y-r}\preceq V_2^\top\nabla_{yy}g(x,y)V_2$. Then we have

$$\begin{aligned}&\|\overline{\nabla}f(x^k,y^{k+1})-\bar{h}_f^k\|\\ \leq&\left\|\nabla h(x^k)^\top A^{\dagger\top}\nabla_{yy}g(x^k,y^{k+1})-\nabla_{xy}g(x^k,y^{k+1})\right\|\|\nabla_yf(x^k,y^{k+1})\|\\ &\times\left\|V_2\left(\left(V_2^\top\nabla_{yy}g(x^k,y^{k+1})V_2\right)^{-1}-\frac{\tilde{c}}{\ell_{g,1}}\sum_{n=0}^{N-1}(I-\frac{\tilde{c}}{\ell_{g,1}}V_2^\top\nabla_{yy}g(x,y)V_2)^n\right)V_2^\top\right\|\end{aligned}$$

$$\leq \frac{\left(1+\ell_{h,0}\|A^\dagger\|\right)\ell_{g,1}\ell_{f,0}\left(1-\frac{\tilde{c}\mu_g}{\ell_{g,1}}\right)^N}{\mu_g} = L_y\ell_{f,0}\left(1-\frac{\tilde{c}\mu_g}{\ell_{g,1}}\right)^N \tag{53}$$

where the second term of (a) is derived from $\|X^{-1}-Y^{-1}\| \leq \|X^{-1}\|\|X-Y\|\|Y^{-1}\|$. Then plugging in the choice of $\tilde{c}$ to (53) results in (48).

The proof of (49) is based on Lemma 18. For ease of narration, we denote

$$\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)}) := \nabla h(x^k)^\top A^{\dagger\top}\nabla_{yy}g(x^k,y^{k+1};\phi^k_{(0)}) - \nabla_{xy}g(x^k,y^{k+1};\phi^k_{(0)})$$
$$\nabla^h_{xy}g(x^k,y^{k+1}) := \nabla h(x^k)^\top A^{\dagger\top}\nabla^2_{yy}g(x^k,y^{k+1}) - \nabla^2_{xy}g(x^k,y^{k+1}).$$

We notice that

$$\mathbb{E}\left[\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})|\mathcal{F}^S_k\right] = \nabla^h_{xy}g(x^k,y^{k+1})$$

and then the bias and variance of $\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})$ can be bounded by

$$\|\nabla^h_{xy}g(x^k,y^{k+1})\|^2 \leq 2(1+\ell^2_{h,0}\|A^\dagger\|^2)\ell^2_{g,1} \tag{54}$$

$$\begin{aligned}
&\mathbb{E}\left[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)}) - \nabla^h_{xy}g(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\leq \|\nabla h(x^k)^\top A^{\dagger\top}\|^2\mathbb{E}\left[\|\nabla_{yy}g(x^k,y^{k+1};\phi^k_{(0)}) - \nabla_{yy}g(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\quad + \mathbb{E}\left[\|\nabla_{xy}g(x^k,y^{k+1}\phi^k_{(0)}) - \nabla_{xy}g(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\leq (1+\ell^2_{h,0}\|A^\dagger\|^2)\sigma^2_{g,2}.
\end{aligned} \tag{55}$$

Thus adding (54) and (55), we arrive at the second moment bound for $\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})$ as

$$\begin{aligned}
&\mathbb{E}\left[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})\|^2|\mathcal{F}^S_k\right]\\
&= \|\mathbb{E}\left[\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})|\mathcal{F}^S_k\right]\|^2 + \mathbb{E}\left[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)}) - \nabla^h_{xy}g(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\leq (1+\ell^2_{h,0}\|A^\dagger\|^2)(2\ell^2_{g,1}+\sigma^2_{g,2}).
\end{aligned} \tag{56}$$

Then the variance of h^k_f can be decomposed and bounded as

$$\begin{aligned}
&\mathbb{E}\left[\|h^k_f - \bar{h}^k_f\|^2|\mathcal{F}^S_k\right]\\
&\leq \mathbb{E}\left[\|\nabla_x f(x^k,y^{k+1};\xi^k) - \nabla_x f(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\quad + \|\nabla h(x^k)A^\dagger\|\mathbb{E}\left[\|\nabla_y f(x^k,y^{k+1};\xi^k) - \nabla_y f(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\right]\\
&\quad + \mathbb{E}\Big[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})V_2H^k_{yy}V_2^\top\nabla_y f(x^k,y^{k+1};\xi^k)\\
&\quad - \nabla^h_{xy}g(x^k,y^{k+1})V_2\mathbb{E}[H^k_{yy}|\mathcal{F}^S_k]V_2^\top\nabla_y f(x^k,y^{k+1})\|^2|\mathcal{F}^S_k\Big]\\
&\overset{(a)}{\leq} (1+\ell_{h,0}\|A^\dagger\|)\sigma^2_f + 3\mathbb{E}[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})\|^2|\mathcal{F}^S_k]\mathbb{E}[\|H^k_{yy}\|^2|\mathcal{F}^S_k]\mathbb{E}[\|\nabla_y f(x^k,y^{k+1};\xi^k) - \nabla_y f(x^k,y^{k+1})\|^2|\mathcal{F}^S_k]\\
&\quad + 3\mathbb{E}[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})\|^2|\mathcal{F}^S_k]\mathbb{E}[\|\nabla_y f(x^k,y^{k+1};\xi^k)\|^2|\mathcal{F}^S_k]\mathbb{E}[\|H^k_{yy} - \mathbb{E}[H^k_{yy}|\mathcal{F}^S_k]\|^2|\mathcal{F}^S_k]\\
&\quad + 3\mathbb{E}[\|H^k_{yy}\|^2|\mathcal{F}^S_k]\mathbb{E}[\|\nabla_y f(x^k,y^{k+1};\xi^k)\|^2|\mathcal{F}^S_k]\mathbb{E}[\|\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)}) - \nabla^h_{xy}g(x^k,y^{k+1})\|^2|\mathcal{F}^S_k]\\
&\overset{(b)}{\leq} (1+\ell_{h,0}\|A^\dagger\|)\sigma^2_f + \frac{3N(1+\ell^2_{h,0}\|A^\dagger\|^2)(d_y-r)}{\ell_{g,1}(\mu^2_g+\sigma^2_{g,2})}\left[(2\ell^2_{g,1}+\sigma^2_{g,2})\left(2\sigma^2_f+\ell^2_{f,0}\right) + (\sigma^2_f+\ell^2_{f,0})\sigma^2_{g,2}\right]\\
&\leq (1+\ell_{h,0}\|A^\dagger\|)\sigma^2_f + \frac{6N(1+\ell^2_{h,0}\|A^\dagger\|^2)(d_y-r)(\ell^2_{g,1}+\sigma^2_{g,2})\left(2\sigma^2_f+\ell^2_{f,0}\right)}{\ell_{g,1}(\mu^2_g+\sigma^2_{g,2})} =: \tilde{\sigma}^2_f
\end{aligned}$$

where (a) comes from (47) and $(A+B+C)^2 \leq 3(A^2+B^2+C^2)$, and (b) comes from the second moment bound and variance of $\nabla^h_{xy}g(x^k,y^{k+1};\phi^k_{(0)})$, H_{yy} and $\nabla_y f(x^k,y^{k+1};\xi^k)$. □

Lemma 20 (Descent of upper level). *Suppose Assumption 1–2 hold, then the sequence of x_k generated by Algorithm 1 satisfies*

$$\begin{aligned}\mathbb{E}[F(x^{k+1})] - \mathbb{E}[F(x^k)] \leq & -\frac{\alpha}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \alpha L_f^2 \mathbb{E}\left[\|y^*(x^k) - y^{k+1}\|^2\right] + \alpha b_k^2 \\ & - \left(\frac{\alpha}{2} - \frac{L_F\alpha^2}{2}\right)\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] + \frac{L_F\alpha^2\tilde{\sigma}_f^2}{2}\end{aligned} \tag{57}$$

where $P_x = I - B^\dagger B$ is the projection matrix of B and $B^\dagger$ is the Moore-Penrose inverse of B.

Proof. Since $\mathcal{X} = \{x \mid Bx = e\}$, for any x, we have that $\mathrm{Proj}_{\mathcal{X}}(x) = (I - B^\dagger B)x + B^\dagger e$ is a linear operator of x according to (26). Thus, we have

$$\begin{aligned}x^{k+1} &= \mathrm{Proj}_{\mathcal{X}}(x^k - \alpha h_f^k) = (I - B^\dagger B)(x^k - \alpha h_f^k) + B^\dagger e \\ &= (I - B^\dagger B)x^k + B^\dagger e - (I - B^\dagger B)(\alpha h_f^k) \\ &= \mathrm{Proj}_{\mathcal{X}}(x^k) - \alpha(I - B^\dagger B)h_f^k \\ &= x^k - \alpha(I - B^\dagger B)h_f^k\end{aligned} \tag{58}$$

where the last equality is due to $x^k \in \mathcal{X}$.

Taking the expectation of $F(x^{k+1})$ conditioned on $\mathcal{F}_k^S$, we get

$$\begin{aligned}\mathbb{E}\left[F(x^{k+1})|\mathcal{F}_k^S\right] &\overset{(a)}{\leq} F(x^k) + \langle \nabla F(x^k), \mathbb{E}\left[x^{k+1} - x^k|\mathcal{F}_k^S\right]\rangle + \frac{L_F}{2}\mathbb{E}\left[\|x^{k+1} - x^k\|^2|\mathcal{F}_k^S\right] \\ &= F(x^k) - \alpha\langle \nabla F(x^k), (I - B^\dagger B)\bar{h}_f^k\rangle + \frac{L_F}{2}\alpha^2\mathbb{E}\left[\|(I - B^\dagger B)h_f^k\|^2|\mathcal{F}_k^S\right] \\ &\overset{(b)}{\leq} F(x^k) - \frac{\alpha}{2}\|(I - B^\dagger B)\nabla F(x^k)\|^2 + \frac{\alpha}{2}\|(I - B^\dagger B)(\nabla F(x^k) - \bar{h}_f^k)\|^2 \\ &\quad - \left(\frac{\alpha}{2} - \frac{L_F\alpha^2}{2}\right)\|(I - B^\dagger B)\bar{h}_f^k\|^2 + \frac{L_F\alpha^2\tilde{\sigma}_f^2}{2} \\ &\overset{(c)}{\leq} F(x^k) - \frac{\alpha}{2}\|\nabla F(x^k)\|_{P_x}^2 + \frac{\alpha}{2}\|\nabla F(x^k) - \bar{h}_f^k\|^2 \\ &\quad - \left(\frac{\alpha}{2} - \frac{L_F\alpha^2}{2}\right)\|\bar{h}_f^k\|_{P_x}^2 + \frac{L_F\alpha^2\tilde{\sigma}_f^2}{2}\end{aligned} \tag{59}$$

where (a) comes from the smoothness of F, (b) is derived from $2a^\top b = \|a\|^2 + \|b\|^2 - \|a - b\|^2$, $\mathbb{E}[\|X\|^2|Y] = \|\mathbb{E}[X|Y]\|^2 + \mathbb{E}[\|X - \mathbb{E}[X|Y]\|^2|Y]$, $(I - B^\dagger B)^2 = I - B^\dagger B$ and Lemma 19, (c) is due to the definition of $\|\cdot\|_{P_x}$ and $\|I - B^\dagger B\| \leq 1$.

Besides, we decompose the gradient bias term as follows

$$\begin{aligned}\|\nabla F(x^k) - \bar{h}_f^k\|^2 &\leq 2\|\nabla F(x^k) - \overline{\nabla} f(x^k, y^{k+1})\|^2 + 2\|\overline{\nabla} f(x^k, y^{k+1}) - \bar{h}_f^k\|^2 \\ &\leq 2\|\overline{\nabla} f(x^k, y^*(x^k)) - \overline{\nabla} f(x^k, y^{k+1})\|^2 + 2b_k^2 \\ &\leq 2L_f^2\|y^*(x^k) - y^{k+1}\| + 2b_k^2.\end{aligned} \tag{60}$$

Plugging (60) to (59) and taking expectation, we get that

$$\begin{aligned}\mathbb{E}[F(x^{k+1})] - \mathbb{E}[F(x^k)] \leq & -\frac{\alpha}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \alpha L_f^2 \mathbb{E}\left[\|y^*(x^k) - y^{k+1}\|^2\right] + \alpha b_k^2 \\ & - \left(\frac{\alpha}{2} - \frac{L_F\alpha^2}{2}\right)\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] + \frac{L_F\alpha^2\tilde{\sigma}_f^2}{2}.\end{aligned}$$

□

Lemma 21 (Error of lower-level update). *Suppose that Assumption 1–2 hold and $\beta \leq \frac{1}{\ell_{g,1}}$, then the error of lower-level variable can be bounded by*

$$\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2] \leq (1-\beta\mu_g)^S \, \mathbb{E}[\|y^k - y^*(x^k)\|^2] + S\beta^2\sigma_{g,1}^2 \tag{61a}$$

$$\begin{aligned}\mathbb{E}[\|y^{k+1} - y^*(x^{k+1})\|^2] \leq &\left(1+\gamma+\eta L_{yx}\widetilde{C}_f^2\alpha^2\right)\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2] \\ &+ \left(L_y^2 + \frac{L_{yx}}{\eta}\right)\alpha^2\tilde{\sigma}_f^2 + \left(L_y^2 + \frac{L_{yx}}{\eta} + \frac{L_y^2}{\gamma}\right)\alpha^2\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right]\end{aligned} \tag{61b}$$

where $\widetilde{C}_f^2 := 2b_k^2 + 2\ell_{f,0}^2\left(1+\frac{\ell_{g,1}}{\mu_g}\right)^2 + \tilde{\sigma}_f^2$, γ and η are balancing constants will be chosen in the final theorem.

Proof. First, since the lower-level objective function is strongly-convex and smooth, when $0 \leq \beta \leq \frac{1}{\ell_{g,1}}$, we have the following fact

$$\begin{aligned}&\|y_1 - \beta\nabla_y g(x,y_1) - (y_2 - \beta\nabla_y g(x,y_2))\|^2 \\ &= \|y_1 - y_2\|^2 + \beta^2\|\nabla_y g(x,y_1) - \nabla_y g(x,y_2))\|^2 - 2\beta\langle y_1 - y_2, \nabla_y g(x,y_1) - \nabla_y g(x,y_2)\rangle \\ &\leq (1-\beta\mu_g)\|y_1 - y_2\|^2 - 2\beta D_g((x,y_1),(x,y_2)) + \beta^2\|\nabla_y g(x,y_1) - \nabla_y g(x,y_2))\|^2 \\ &\leq (1-\beta\mu_g)\|y_1 - y_2\|^2\end{aligned} \tag{62}$$

where the first inequality is according to (24) and the last inequality is due to (25) and $\beta \leq \frac{1}{\ell_{g,1}}$.

Then, for each lower-level update, we obtain that

$$\begin{aligned}&\mathbb{E}[\|y^{k,s+1} - y^*(x^k)\|^2|\mathcal{F}_{k,s}] \\ &= \mathbb{E}[\|\operatorname{Proj}_{\mathcal{Y}(x^k)}(y^{k,s} - \beta\nabla_y g(x^k, y^{k,s}; \phi^{k,s})) - \operatorname{Proj}_{\mathcal{Y}(x^k)}(y^*(x^k) - \beta\nabla_y g(x^k, y^*(x^k)))\|^2|\mathcal{F}_{k,s}] \\ &\leq \mathbb{E}[\|y^{k,s} - \beta\nabla_y g(x^k, y^{k,s}; \phi^{k,s}) - y^*(x^k) + \beta\nabla_y g(x^k, y^*(x^k))\|^2|\mathcal{F}_{k,s}] \\ &\leq \mathbb{E}[\|y^{k,s} - \beta\nabla_y g(x^k, y^{k,s}) - y^*(x^k) + \beta\nabla_y g(x^k, y^*(x^k))\|^2|\mathcal{F}_{k,s}] \\ &\quad + \beta^2\mathbb{E}[\|\nabla_y g(x^k, y^{k,s}; \phi^{k,s}) - \nabla_y g(x^k, y^{k,s})\|^2|\mathcal{F}_{k,s}] \\ &\leq (1-\beta\mu_g)\|y^{k,s} - y^*(x^k)\|^2 + \beta^2\sigma_{g,1}^2\end{aligned} \tag{63}$$

where the first inequality is due to $y^*(x^k) = \operatorname{Proj}_{\mathcal{Y}(x^k)}(y^*(x^k) - \beta\nabla_y g(x^k, y^*(x^k)))$ and the last inequality is obtained by (62) with $x = x^k, y_2 = y^*(x^k), y_1 = y^{k,s}$, and Assumption 3. Taking expectation of both sides in (63), one have

$$\mathbb{E}[\|y^{k,s+1} - y^*(x^k)\|^2] \leq (1-\beta\mu_g)\mathbb{E}[\|y^{k,s} - y^*(x^k)\|^2] + \beta^2\sigma_{g,1}^2. \tag{64}$$

Thus, (61a) can be obtained by telescoping (64).

On the other hand, we have

$$\begin{aligned}\|y^{k+1} - y^*(x^{k+1})\|^2 = \|y^{k+1} - y^*(x^k)\|^2 + \underbrace{\|y^*(x^k) - y^*(x^{k+1})\|^2}_{J_1} \\ + 2\underbrace{\langle y^{k+1} - y^*(x^k), y^*(x^k) - y^*(x^{k+1})\rangle}_{J_2}.\end{aligned}$$

Since $y^*(x)$ is L_y Lipschitz continuous, J_1 can be bounded by

$$\begin{aligned}\mathbb{E}[J_1] &\leq L_y^2\mathbb{E}\left[\|x^{k+1} - x^k\|^2\right] \\ &\overset{(a)}{=} \alpha^2 L_y^2\mathbb{E}\left[\mathbb{E}\left[\|(I - B^\dagger B)h_f^k\|^2\right]|\mathcal{F}_k^S\right] \\ &\overset{(b)}{\leq} \alpha^2 L_y^2\left(\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] + \tilde{\sigma}_f^2\right)\end{aligned} \tag{65}$$

where (a) comes from (58), (b) holds since $\mathbb{E}[\|C\|^2|D] = \|\mathbb{E}[C|D]\|^2 + \mathbb{E}[\|C - \mathbb{E}[C|D]\|^2|D]$ and Lemma 19.

On the other hand, we can decompose J_2 by two terms as follows.

$$J_2 = \underbrace{-\langle y^{k+1} - y^*(x^k), \nabla y^*(x^k)^\top (x^{k+1} - x^k)\rangle}_{J_{2,1}} \underbrace{-\langle y^{k+1} - y^*(x^k), y^*(x^{k+1}) - y^*(x^k) - \nabla y^*(x^k)^\top (x^{k+1} - x^k)\rangle}_{J_{2,2}}.$$

Moreover, the conditional expectation of $J_{2,1}$ can be bounded by

$$\begin{aligned}\mathbb{E}[J_{2,1}|\mathcal{F}_k^S] &= -\langle y^{k+1} - y^*(x^k), \mathbb{E}[\nabla y^*(x^k)^\top (x^{k+1} - x^k)|\mathcal{F}_k^S]\rangle \\ &\le -\alpha\langle y^{k+1} - y^*(x^k), \nabla y^*(x^k)^\top (I - B^\dagger B)\bar{h}_f^k\rangle \\ &\overset{(a)}{\le} \frac{\gamma}{2}\|y^{k+1} - y^*(x^k)\|^2 + \frac{\alpha^2 L_y^2}{2\gamma}\|\bar{h}_f^k\|_{P_x}^2 \end{aligned} \tag{66}$$

where (a) comes form Young's inequality and the boundedness of $\nabla y^*(x^k)$. Then taking expectation of (66), we obtain that

$$\mathbb{E}[J_{2,1}] \le \frac{\gamma}{2}\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2] + \frac{\alpha^2 L_y^2}{2\gamma}\mathbb{E}[\|\bar{h}_f^k\|_{P_x}^2]. \tag{67}$$

Based on the smoothness of $y^*(x)$ and Jensen inequality, $J_{2,2}$ can be bounded by

$$\begin{aligned}\mathbb{E}[J_{2,2}] &\le \mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|\|y^*(x^{k+1}) - y^*(x^k) - \nabla y^*(x^k)^\top (x^{k+1} - x^k)\|^2\right] \\ &\le \frac{L_{yx}}{2}\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|\|x^{k+1} - x^k\|^2\right] \\ &\overset{(a)}{\le} \frac{\eta L_{yx}\alpha^2}{2}\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2\|h_f^k\|^2] + \frac{L_{yx}\alpha^2}{2\eta}\mathbb{E}\left[\|h_f^k\|_{P_x}^2\right] \\ &\le \frac{\eta L_{yx}\alpha^2}{2}\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2\mathbb{E}[\|h_f^k\|^2|\mathcal{F}_k^S]\right] + \frac{L_{yx}\alpha^2}{2\eta}\mathbb{E}\left[\mathbb{E}\left[\|h_f^k\|_{P_x}^2|\mathcal{F}_k^S\right]\right] \\ &\le \frac{\eta L_{yx}\alpha^2}{2}\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2\left(\|\bar{h}_f^k\|^2 + \tilde{\sigma}_f^2\right)\right] + \frac{L_{yx}\alpha^2}{2\eta}\left(\mathbb{E}[\|\bar{h}_f^k\|_{P_x}^2] + \tilde{\sigma}_f^2\right) \\ &\le \frac{\eta L_{yx}\alpha^2}{2}\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2\left(2\|\bar{h}_f^k - \overline{\nabla} f(x^k, y^{k+1})\|^2 + 2\|\overline{\nabla} f(x^k, y^{k+1})\|^2 + \tilde{\sigma}_f^2\right)\right] \\ &\quad + \frac{L_{yx}\alpha^2}{2\eta}\left(\mathbb{E}[\|\bar{h}_f^k\|_{P_x}^2] + \tilde{\sigma}_f^2\right) \\ &\overset{(b)}{\le} \frac{\eta L_{yx}\alpha^2}{2}\left(2b_k^2 + 2\ell_{f,0}^2\left(1 + L_y\right)^2 + \tilde{\sigma}_f^2\right)\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2\right] \\ &\quad + \frac{L_{yx}\alpha^2}{2\eta}\left(\mathbb{E}[\|\bar{h}_f^k\|_{P_x}^2] + \tilde{\sigma}_f^2\right). \end{aligned} \tag{68}$$

where (a) comes from the update (58), Young's inequality and $\|h_f^k\|_{P_x} = \|(I - B^\dagger B)h_f^k\| \le \|h_f^k\|$ and (b) holds from Lemma 15 and Lemma 19. Then denoting $\widetilde{C}_f^2 := 2b_k^2 + 2\ell_{f,0}^2\left(1 + L_y\right)^2 + \tilde{\sigma}_f^2$ and combining (65), (67) and (68), we get

$$\begin{aligned}\mathbb{E}[\|y^{k+1} - y^*(x^{k+1})\|^2] \le \left(1 + \gamma + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right)\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2] + \left(L_y^2 + \frac{L_{yx}}{\eta}\right)\alpha^2\tilde{\sigma}_f^2 \\ + \left(L_y^2 + \frac{L_{yx}}{\eta} + \frac{L_y^2}{\gamma}\right)\alpha^2\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right].\end{aligned}$$

This completes the proof. □

C.4 Proof of Theorem 4

We first restate a formal version of Theorem 4 as follows.

Restatement of Theorem 4. *Under Assumption 1–4, defining the constants as*

$$\bar{\alpha}_1 = \frac{1}{2L_F + 4L_f L_y + \frac{4L_f L_{yx}}{\eta L_y}}, \qquad \bar{\alpha}_2 = \frac{\mu_g}{\ell_{g,1}(5L_f L_y + \eta L_{yx}\tilde{C}_f^2)} \tag{69}$$

and choosing

$$\alpha = \min\left(\bar{\alpha}_1, \bar{\alpha}_2, \frac{\bar{\alpha}}{\sqrt{K}}\right), \qquad \beta = \frac{5L_f L_y + \eta L_{yx}\tilde{C}_f^2}{\mu_g}\alpha, \qquad N = \mathcal{O}(\log K)$$

then for any $S \geq 1$ in Algorithm 1, we have

$$\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] = \mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$$

where $P_x = I - B^\dagger B$ is the projection matrix on $\mathrm{Ker}(B)$ and $\|x\|_{P_x} = \sqrt{x^\top P_x x}$ is the weighted Euclidean norm associated with P_x.

Proof. According to Lemma 21 and plugging (61b) into (61a), we get that

$$\begin{aligned}
\mathbb{E}[\|y^{k+1} - y^*(x^{k+1})\|^2] \leq{}& \left(1 + \gamma + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right)(1-\beta\mu_g)^S\,\mathbb{E}[\|y^k - y^*(x^k)\|^2] \\
&+ \left(1 + \gamma + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right) S\beta^2\sigma_{g,1}^2 + \left(L_y^2 + \frac{L_{yx}}{\eta}\right)\alpha^2\tilde{\sigma}_f^2 \\
&+ \left(L_y^2 + \frac{L_{yx}}{\eta} + \frac{L_y^2}{\gamma}\right)\alpha^2\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right].
\end{aligned} \tag{70}$$

We can define Lyapunov function as

$$\mathbb{V}^k := F(x^k) + \frac{L_f}{L_y}\|y^*(x^k) - y^k\|^2$$

Using Lemma 19–21, we get

$$\begin{aligned}
\mathbb{E}\left[\mathbb{V}^{k+1}\right] - \mathbb{E}\left[\mathbb{V}^k\right] \leq{}& -\frac{\alpha}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \alpha L_f^2\,(1-\beta\mu_g)^S\,\mathbb{E}\left[\|y^k - y^*(x^k)\|^2\right] + \alpha b_k^2 \\
&- \left(\frac{\alpha}{2} - \frac{L_F\alpha^2}{2}\right)\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] + \frac{L_F\alpha_k^2\tilde{\sigma}_f^2}{2} \\
&+ \frac{L_f}{L_y}\left[\left(1 + \gamma + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right)(1-\beta\mu_g)^S - 1\right]\mathbb{E}\left[\|y^k - y^*(x^k)\|^2\right] \\
&+ \frac{L_f}{L_y}\left(1 + \gamma + L_y L_f\alpha + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right) S\beta^2\sigma_{g,1}^2 \\
&+ \frac{L_f}{L_y}\left(L_y^2 + \frac{L_{yx}}{\eta}\right)\alpha^2\tilde{\sigma}_f^2 + \frac{L_f}{L_y}\left(L_y^2 + \frac{L_{yx}}{\eta} + \frac{L_y^2}{\gamma}\right)\alpha^2\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] \\
\leq{}& -\frac{\alpha}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \frac{L_f}{L_y}\left(1 + \gamma + L_y L_f\alpha + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right)\beta^2 S\sigma_{g,1}^2 \\
&+ \alpha b_k^2 + \left[\frac{L_F}{2} + \frac{L_f}{L_y}\left(L_y^2 + \frac{L_{yx}}{\eta}\right)\right]\alpha^2\tilde{\sigma}_f^2 \\
&- \left[\frac{\alpha}{2} - \left(\frac{L_F}{2} + L_f L_y\left(1 + \frac{1}{\gamma}\right) + \frac{L_f L_{yx}}{\eta L_y}\right)\alpha^2\right]\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right] \\
&- \left(\frac{L_f\mu_g\beta}{L_y} - \alpha L_f^2 - \frac{L_f\gamma}{L_y} - \frac{\eta L_f L_{yx}\tilde{C}_f^2\alpha^2}{L_y}\right)\mathbb{E}\left[\|y^k - y^*(x^k)\|^2\right]
\end{aligned} \tag{71}$$

Selecting $\gamma = 4L_f L_y\alpha$, (71) can be simplified by

$$\mathbb{E}[\mathbb{V}^{k+1}] - \mathbb{E}[\mathbb{V}^k] \leq -\frac{\alpha}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \frac{L_f}{L_y}\left(1 + 5L_f L_y\alpha + \eta L_{yx}\widetilde{C}_f^2\alpha^2\right)\beta^2 S\sigma_{g,1}^2$$

$$+\alpha b_k^2+\left[\frac{L_F}{2}+\frac{L_f}{L_y}\left(L_y^2+\frac{L_{yx}}{\eta}\right)\right]\alpha^2\tilde{\sigma}_f^2$$
$$-\left[\frac{\alpha}{4}-\left(\frac{L_F}{2}+L_fL_y+\frac{L_fL_{yx}}{\eta L_y}\right)\alpha^2\right]\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right]$$
$$-\left(\frac{L_f\mu_g\beta}{L_y}-5\alpha L_f^2-\frac{\eta L_fL_{yx}\tilde{C}_f^2\alpha^2}{L_y}\right)\mathbb{E}\left[\|y^k-y^*(x^k)\|^2\right] \tag{72}$$

Let $\alpha\le 1$, then the sufficient condition of making the last two terms negative becomes

$$\alpha\le\min\left(\frac{1}{2L_F+4L_fL_y+\frac{4L_f\mu_g\ell_{g,1}}{L_y}},\frac{\mu_g\beta}{5L_fL_y+\eta L_{yx}\tilde{C}_f^2}\right). \tag{73}$$

Since we also need $\beta\le\frac{1}{\ell_{g,1}}$, then the sufficient condition for (73) becomes

$$\alpha\le\min\left(\frac{1}{2L_F+4L_fL_y+\frac{4L_f\mu_g\ell_{g,1}}{L_y}},\frac{\mu_g}{\ell_{g,1}(5L_fL_y+\eta L_{yx}\tilde{C}_f^2)}\right),\beta=\frac{5L_fL_y+\eta L_{yx}\tilde{C}_f^2}{\mu_g}\alpha.$$

Denoting

$$\bar{\alpha}_1=\frac{1}{2L_F+4L_fL_y+\frac{4L_fL_{yx}}{\eta L_y}},\qquad \bar{\alpha}_2=\frac{\mu_g}{\ell_{g,1}(5L_fL_y+\eta L_{yx}\tilde{C}_f^2)}$$

and choosing $\alpha=\min\left(\bar{\alpha}_1,\bar{\alpha}_2,\frac{\bar{\alpha}}{\sqrt{K}}\right),\beta=\frac{5L_fL_y+\eta L_{yx}\tilde{C}_f^2}{\mu_g}\alpha$, then (72) becomes

$$\frac{\alpha}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2]\le\left(\mathbb{E}[\mathbb{V}^k]-\mathbb{E}[\mathbb{V}^{k+1}]\right)+c_1S\alpha^2\sigma_{g,1}^2+c_2\alpha^2\tilde{\sigma}_f^2+\alpha b_k^2 \tag{74}$$

where c_1 and c_2 are defined as

$$c_1=\frac{L_f}{L_y}\left(1+5L_fL_y\alpha+\eta L_{yx}\widetilde{C}_f^2\alpha^2\right)\left(\frac{5L_fL_y+\eta L_{yx}\tilde{C}_f^2}{\mu_g\beta}\right)^2$$
$$c_2=\frac{L_F}{2}+\frac{L_f}{L_y}\left(L_y^2+\frac{L_{yx}}{\eta}\right).$$

Telescoping (74) and dividing both sides by $\frac{1}{2}\sum_{k=0}^{K-1}\alpha$ leads to

$$\frac{\sum_{k=0}^{K-1}\alpha\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right]}{\sum_{k=0}^{K-1}\alpha}\le\frac{\mathbb{V}^0+\sum_{k=0}^{K-1}\alpha b_k^2+c_1S\alpha\sigma_{g,1}^2+c_2\alpha\tilde{\sigma}_f^2}{\frac{1}{2}\sum_{k=0}^{K-1}\alpha}$$

Let $\bar{\alpha},S=\mathcal{O}(1)$ and $N=\mathcal{O}(\log K)$, then we know $\tilde{\sigma}_f^2=\mathcal{O}(N)=\mathcal{O}(\log K)$, and thus,

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right]\le\frac{\mathbb{V}^0+\sum_{k=0}^{K-1}\alpha b_k^2+c_1S\alpha\sigma_{g,1}^2+c_2\alpha\tilde{\sigma}_f^2}{\frac{1}{2}\sum_{k=0}^{K-1}\alpha}=\mathcal{O}\left(\frac{\log(K)}{\sqrt{K}}\right)=\tilde{\mathcal{O}}\left(\frac{1}{\sqrt{K}}\right).$$

Therefore, Algorithm 1 achieves an ϵ-stationary point by $\tilde{\mathcal{O}}(\epsilon^{-2})$ iterations, which matches the iteration complexity of single level gradient descent method. □

D Proof of Algorithm 2

In this section, we present the proof of Algorithm 2. We define

$$\widetilde{\mathcal{F}}_k^t:=\sigma\{y^0,x^0,\cdots,y^{k+1},x^{k,1},\cdots,x^{k,t}\} \tag{75}$$

where $\sigma\{\cdot\}$ denotes the σ-algebra generated by the random variables. Then it follows that $\widetilde{\mathcal{F}}_k^0=\sigma\{y^0,x^0,\cdots,y^{k+1}\}$.

D.1 Descent of upper level

First, we notice that the update of x^{k+1} can be written as

$$\begin{aligned} x^{k+1} &\overset{(a)}{=} (1-\delta)x^k + \delta \operatorname{Proj}_{\mathcal{X}}\left(x^k - \alpha \sum_{t=0}^{T-1} h_f^{k,t}\right) \\ &\overset{(b)}{=} (1-\delta)x^k + \delta(I - B^\dagger B)\left(x^k - \alpha \sum_{t=0}^{T-1} h_f^{k,t}\right) + \delta B^\dagger e \\ &= (1-\delta)x^k + \delta \operatorname{Proj}_{\mathcal{X}}(x^k) - \alpha\delta \sum_{t=0}^{T-1}(I - B^\dagger B) h_f^{k,t} \\ &\overset{(c)}{=} x^k - \alpha\delta(I - B^\dagger B)\left(\sum_{t=0}^{T-1} h_f^{k,t}\right) \end{aligned} \tag{76}$$

where (a) comes from the update rule in 2, (b) is derived from the closed form of $\operatorname{Proj}(\cdot)$ on the linear space $\mathcal{X} = \{x \mid Bx = e\}$, and (c) holds since $x^k \in \mathcal{X}$.

We first quantify the bias induced by querying Hessian inverse vector product at different point.

Lemma 22 (Error of using delayed Hessian vector product). *Define*

$$\begin{aligned} G(x,y,\tilde{x}) := \nabla_x f(x,y) + \big[&\left(\nabla h(\tilde{x})^\top A^{\dagger\top} \nabla_{yy} g(\tilde{x},y) - \nabla_{xy} g(\tilde{x},y)\right) \\ &\times V_2 (V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} V_2^\top - \nabla h(\tilde{x})^\top A^{\dagger\top}\big] \nabla_y f(\tilde{x},y) \end{aligned}$$

as the gradient estimator using Hessian vector product at point $\tilde{x}$. We have

$$\begin{aligned} &\|G(x,y,\tilde{x})\| \le \ell_{f,0}(1+L_y) \\ &\|G(x,y,\tilde{x}) - \overline{\nabla} f(x,y)\| \le L_G \|x - \tilde{x}\| \end{aligned}$$

where $L_G := \frac{\left(1+\ell_{h,0}\|A^\dagger\|\right)\ell_{g,1}}{\mu_g}\left(\ell_{f,1} + \frac{\ell_{f,0}\ell_{g,2}}{\mu_g} + \frac{\mu_g \ell_{g,2}}{\ell_{g,1}}\right) + \left(\ell_{h,0}\ell_{f,1} + \ell_{h,1}(\ell_{g,1}+\ell_{f,0})\right)\|A^\dagger\|$.

Proof. First, since $G(x,y,\tilde{x})$ only differs from $\overline{\nabla} f(x,y)$ at the evaluation point $\tilde{x}$ of Hessian vector product, the bound for $\|G(x,y,\tilde{x})\|$ can derived the same way as $\overline{\nabla} f(x,y)$ following Lemma 15, that is

$$\|G(x,y,\tilde{x})\| \le \ell_{f,0}(1+L_y).$$

Next, for any $x, y, \tilde{x}$, we have that

$$\begin{aligned} &\|G(x,y,\tilde{x}) - \overline{\nabla} f(x,y)\| \\ &\quad\le \| \left(\nabla h(\tilde{x})^\top A^{\dagger\top} \nabla_{yy} g(\tilde{x},y) - \nabla_{xy} g(\tilde{x},y)\right) V_2 (V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} V_2^\top \nabla_y f(\tilde{x},y) \\ &\qquad - \left(\nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y) - \nabla_{xy} g(x,y)\right) V_2 (V_2^\top \nabla_{yy} g(x,y) V_2)^{-1} V_2^\top \nabla_y f(x,y)\| \\ &\qquad + \|\nabla h(\tilde{x})^\top A^{\dagger\top} \nabla_y f(\tilde{x},y) - \nabla h(x)^\top A^{\dagger\top} \nabla_y f(x,y)\| \\ &\quad\le \|\nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y) - \nabla_{xy} g(x,y)\| \\ &\qquad \times \|V_2 (V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} V_2^\top \nabla_y f(\tilde{x},y) - V_2 (V_2^\top \nabla_{yy} g(x,y) V_2)^{-1} V_2^\top \nabla_y f(x,y)\| \\ &\qquad + \|V_2 (V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} V_2^\top \nabla_y f(\tilde{x},y)\| \\ &\qquad \times \|\nabla h(\tilde{x})^\top A^{\dagger\top} \nabla_{yy} g(\tilde{x},y) - \nabla_{xy} g(\tilde{x},y) - \nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y) + \nabla_{xy} g(x,y)\| \\ &\qquad + \|\nabla h(\tilde{x})^\top A^{\dagger\top}\| \|\nabla_y f(\tilde{x},y) - \nabla_y f(x,y)\| + \|\nabla h(\tilde{x})^\top A^{\dagger\top} - \nabla h(x)^\top A^{\dagger\top}\| \|\nabla_y f(x,y)\| \\ &\quad\overset{(a)}{\le} \ell_{g,1}\left(1 + \ell_{h,0}\|A^\dagger\|\right)\left(\|V_2 (V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} V_2^\top\| \|\nabla_y f(\tilde{x},y) - \nabla_y f(x,y)\|\right. \\ &\qquad + \|\nabla_y f(x,y)\| \|V_2((V_2^\top \nabla_{yy} g(\tilde{x},y) V_2)^{-1} - (V_2^\top \nabla_{yy} g(x,y) V_2)^{-1}) V_2^\top\|\big) \\ &\qquad + \|\nabla_{xy} g(\tilde{x},y) - \nabla_{xy} g(x,y)\| + \|\nabla h(\tilde{x})^\top A^{\dagger\top} \nabla_{yy} g(\tilde{x},y) - \nabla h(x)^\top A^{\dagger\top} \nabla_{yy} g(x,y)\| \end{aligned}$$

$$
\begin{aligned}
&+ \ell_{h,0}\ell_{f,1}\|A^\dagger\|\|\tilde{x}-x\| + \ell_{h,1}\ell_{f,0}\|A^\dagger\|\|x-\tilde{x}\| \\
&\overset{(b)}{\leq} \left[\frac{\left(1+\ell_{h,0}\|A^\dagger\|\right)\ell_{g,1}}{\mu_g}\left(\ell_{f,1}+\frac{\ell_{f,0}\ell_{g,2}}{\mu_g}+\frac{\mu_g\ell_{g,2}}{\ell_{g,1}}\right)+\left(\ell_{h,0}\ell_{f,1}+\ell_{h,1}(\ell_{g,1}+\ell_{f,0})\right)\|A^\dagger\|\right]\|x-\tilde{x}\|
\end{aligned}
$$

where (a) and (b) hold similarly with the derivation of (36). □

Lemma 22 shows the bias induced by evaluating the Hessian inverse vector product at different point can be controlled by the point difference.

Then we have the following lemma which is a counterpart of Lemma 19.

Lemma 23 (Bias and variance of gradient estimator). *Let* $\tilde{c}=\frac{\mu_g}{\mu_g^2+\sigma_{g,2}^2}$ *and define*

$$\bar{h}_f^{k,t} := \mathbb{E}[h_f^{k,t}|\widetilde{\mathcal{F}}_k^t],$$

then $h_f^{k,t}$ *is a biased estimator of upper level gradient which satisfies that*

$$\|\bar{h}_f^{k,t} - G(x^{k,t}, y^{k+1}, x^k)\| \leq b_k \tag{77}$$

$$\mathbb{E}\left[\|h_f^{k,t}-\bar{h}_f^{k,t}\|^2|\widetilde{\mathcal{F}}_k^t\right] \leq \tilde{\sigma}_f^2 = \mathcal{O}\left(N\kappa^2\right) \tag{78}$$

$$\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}(h_f^{k,t}-\bar{h}_f^{k,t})\right\|^2|\widetilde{\mathcal{F}}_k^0\right] \leq \frac{\tilde{\sigma}_f^2}{T}. \tag{79}$$

Proof. We omit the proof of (77) and (78) since they are almost the same with the proof of Lemma 19, and only prove (79). For (79), we have

$$
\begin{aligned}
\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}(h_f^{k,t}-\bar{h}_f^{k,t})\right\|^2|\widetilde{\mathcal{F}}_k^0\right] &= \frac{1}{T^2}\mathbb{E}\left[\mathbb{E}\left[\left\|\sum_{t=0}^{T-1}h_f^{k,t}-\bar{h}_f^{k,t}\right\|^2|\widetilde{\mathcal{F}}_k^{T-1}\right]|\widetilde{\mathcal{F}}_k^0\right] \\
&\overset{(a)}{=} \frac{1}{T^2}\mathbb{E}\left[\mathbb{E}\left[\|h_f^{k,T-1}-\bar{h}_f^{k,T-1}\|^2+\left\|\sum_{t=0}^{T-2}h_f^{k,t}-\bar{h}_f^{k,t}\right\|^2|\widetilde{\mathcal{F}}_k^{T-1}\right]|\widetilde{\mathcal{F}}_k^0\right] \\
&= \frac{1}{T^2}\mathbb{E}\left[\|h_f^{k,T-1}-\bar{h}_f^{k,T-1}\|^2|\widetilde{\mathcal{F}}_k^0\right]+\frac{1}{T^2}\mathbb{E}\left[\left\|\sum_{t=0}^{T-2}h_f^{k,t}-\bar{h}_f^{k,t}\right\|^2|\widetilde{\mathcal{F}}_k^0\right] \\
&\overset{(b)}{=} \frac{1}{T^2}\sum_{t=0}^{T-1}\mathbb{E}\left[\|h_f^{k,t}-\bar{h}_f^{k,t}\|^2|\widetilde{\mathcal{F}}_k^0\right] \leq \frac{\tilde{\sigma}_f^2}{T}
\end{aligned}
$$

where (a) is due to $h_f^{k,T-1}-\bar{h}_f^{k,T-1}$ is independent with $\sum_{t=0}^{T-2}h_f^{k,t}-\bar{h}_f^{k,t}$ when given $\mathcal{F}_{k,T-1}^\star$ and (b) follows from applying the previous procedure $T-1$ times. □

We then state a lemma controlling the drifting error of lazy projections.

Lemma 24 (Drifting error of upper level). *Under Assumption 1–4, it holds that for any* t*,*

$$\mathbb{E}\left[\|x^{k,t}-x^k\|^2\right] \leq \alpha^2 t^2 \tilde{C}_f^2$$

where $\widetilde{C}_f^2 = 2b_k^2 + 2\ell_{f,0}^2\left(1+L_y\right)^2 + \tilde{\sigma}_f^2$.

Proof. For any t, we know that

$$\mathbb{E}\left[\|x^{k,t+1}-x^{k,t}\|^2\right] = \mathbb{E}\left[\|x^{k,t+1}-x^{k,t}\|^2\right] = \mathbb{E}\left[\|x^{k,t}-\alpha h_f^{k,t}-x^{k,t}\|^2\right]$$

$$= \alpha^2 \mathbb{E}\left[\|h_f^{k,t}\|^2\right] \overset{(a)}{\leq} \alpha^2 \tilde{C}_f^2. \tag{80}$$

where $\widetilde{C}_f^2 = 2b_k^2 + 2\ell_{f,0}^2 \left(1 + L_y\right)^2 + \tilde{\sigma}_f^2$ and (a) is derived similarly from the bound for $\mathbb{E}\left[\|h_f^k\|^2\right]$ in (68) based on Lemma 22 and Lemma 23.

Then for any t, it holds that

$$\begin{aligned}
\mathbb{E}\left[\|x^{k,t} - x^k\|^2\right] &= \mathbb{E}\left[\|x^{k,t} - x^{k,t-1} + x^{k,t-1} - x^{k,t-2} + \cdots - x^k\|^2\right] \\
&\leq t \sum_{\tau=0}^{t-1} \mathbb{E}\left[\|x^{k,\tau+1} - x^{k,\tau}\|^2\right] \\
&\leq \alpha^2 t^2 \tilde{C}_f^2
\end{aligned}$$

which completes the proof. □

Lemma 25. *Under Assumption 1–4 and let $N = \mathcal{O}(\log \alpha^{-1})$, it holds that*

$$\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1} \bar{h}_f^{k,t} - \overline{\nabla} f(x^k, y^{k+1})\right\|^2\right] \leq \mathcal{O}\left(\alpha^2 T^2\right). \tag{81}$$

Proof. For any t, we have that

$$\begin{aligned}
&\mathbb{E}\left[\|\bar{h}_f^{k,t} - \overline{\nabla} f(x^k, y^{k+1})\|^2\right] \\
&\leq 3\mathbb{E}[\|\bar{h}_f^{k,t} - G(x^{k,t}, y^{k+1}, x^k)\|^2] + 3\mathbb{E}[\|G(x^{k,t}, y^{k+1}, x^k) - \overline{\nabla} f(x^{k,t}, y^{k+1})\|^2] \\
&\quad + 3\mathbb{E}[\|\overline{\nabla} f(x^{k,t}, y^{k+1}) - \overline{\nabla} f(x^k, y^{k+1})\|^2] \\
&\overset{(a)}{\leq} 3b_k^2 + 3L_G^2 \mathbb{E}[\|x^{k,t} - x^k\|^2] + 3L_f^2 \mathbb{E}[\|x^{k,t} - x^k\|^2] \\
&\overset{(b)}{\leq} 3b_k^2 + 3\alpha^2 (L_G^2 + L_f^2) t^2 \tilde{C}_f^2 \overset{(c)}{\leq} \mathcal{O}(\alpha^2 t^2)
\end{aligned} \tag{82}$$

where (a) is due to Lemma 22, (b) comes from Lemma 24 and (c) holds by (48) and $N = \mathcal{O}(\log \alpha^{-1})$. Thus, we have

$$\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1} \bar{h}_f^{k,t} - \overline{\nabla} f(x^k, y^{k+1})\right\|^2\right] \leq \mathcal{O}\left(\alpha^2 T^2\right). \tag{83}$$

which results from (82) and the fact that for any vector sequence $\{z_t\}_{t=0}^{T-1}$,

$$\left\|\frac{1}{T}\sum_{t=0}^{T-1} z_t\right\|^2 = \frac{1}{T^2}\left\|\sum_{t=0}^{T-1} z_t\right\|^2 \leq \frac{1}{T}\sum_{t=0}^{T-1} \|z_t\|^2$$

□

Lemma 26 (Descent of upper level). *Under Assumption 1–4 and define*

$$\bar{\bar{h}}_f^k := \frac{1}{T}\sum_{t=0}^{T-1} \bar{h}_f^{k,t}$$

it holds that

$$\begin{aligned}
\mathbb{E}\left[F(x^{k+1})\right] \leq \mathbb{E}[F(x^k)] - \frac{\alpha T}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2] - \left(\frac{\alpha T}{2} - \frac{\alpha^2 L_F T^2}{2}\right)\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2] + \frac{\alpha^2 L_F T \tilde{\sigma}_f^2}{2} \\
+ \alpha T L_y^2 \mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2\right] + \mathcal{O}(\alpha^3 T^3).
\end{aligned}$$

Proof. First, we have

$$\begin{aligned}\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}h_f^{k,t}\right\|_{P_x}^2\right] &= \mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}h_f^{k,t}-\bar{h}_f^{k,t}+\bar{h}_f^{k,t}\right\|_{P_x}^2\right]\\ &\overset{(a)}{=} \mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}\bar{h}_f^{k,t}\right\|_{P_x}^2\right]+\mathbb{E}\left[\left\|\frac{1}{T}\sum_{t=0}^{T-1}h_f^{k,t}-\bar{h}_f^{k,t}\right\|_{P_x}^2\right]\\ &\overset{(b)}{\le} \mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]+\frac{\tilde{\sigma}_f^2}{T}\end{aligned} \tag{84}$$

where (a) follows from $\mathbb{E}[\|X+Y\|^2]=\mathbb{E}[\|X\|^2]+\mathbb{E}[\|Y\|^2]$ when X and Y are independent and (b) results from (79). Moreover, it follows that

$$\begin{aligned}\mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)\frac{1}{T}\sum_{t=0}^{T-1}h_f^{k,t}\rangle\right] &= \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)h_f^{k,t}\rangle|\widetilde{\mathcal{F}}_k^t\right]\right]\\ &= \mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)\frac{1}{T}\sum_{t=0}^{T-1}\bar{h}_f^{k,t}\rangle\right]\\ &= \mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)\bar{\bar{h}}_f^{k,t}\rangle\right].\end{aligned} \tag{85}$$

Taking the expectation of $F(x^{k+1})$, we get

$$\begin{aligned}\mathbb{E}\left[F(x^{k+1})\right] &\overset{(a)}{\le} \mathbb{E}\left[F(x^k)\right]+\mathbb{E}\left[\langle\nabla F(x^k),x^{k+1}-x^k\rangle\right]+\frac{L_F}{2}\mathbb{E}\left[\|x^{k+1}-x^k\|^2\right]\\ &\overset{(b)}{\le} \mathbb{E}[F(x^k)]-\alpha\delta T\mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)\frac{1}{T}\sum_{t=0}^{T-1}h_f^{k,t}\rangle\right]+\frac{L_F}{2}\alpha^2T^2\delta^2\mathbb{E}\left[\|\frac{1}{T}\sum_{t=0}^{T}h_f^{k,t}\|_{P_x}^2\right]\\ &\overset{(c)}{\le} \mathbb{E}[F(x^k)]-\alpha\delta T\mathbb{E}\left[\langle\nabla F(x^k),(I-B^\dagger B)\bar{\bar{h}}_f^{k,t}\rangle\right]+\frac{L_F}{2}\alpha^2T^2\delta^2\mathbb{E}\left[\|\bar{\bar{h}}_f^{k,t}\|_{P_x}^2\right]+\frac{L_F}{2}\alpha^2T\delta^2\tilde{\sigma}_f^2\\ &\overset{(d)}{\le} \mathbb{E}[F(x^k)]-\frac{\alpha\delta T}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2]-\left(\frac{\alpha\delta T}{2}-\frac{\alpha^2L_FT^2\delta^2}{2}\right)\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]\\ &\quad+\frac{\alpha\delta T}{2}\mathbb{E}\left[\|\nabla F(x^k)-\bar{\bar{h}}_f^{k,t}\|^2\right]+\frac{\alpha^2\delta^2L_FT\tilde{\sigma}_f^2}{2}\\ &\le \mathbb{E}[F(x^k)]-\frac{\alpha\delta T}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2]-\left(\frac{\alpha\delta T}{2}-\frac{\alpha^2\delta^2L_FT^2}{2}\right)\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]+\frac{\alpha^2\delta^2L_FT\tilde{\sigma}_f^2}{2}\\ &\quad+\alpha\delta T\mathbb{E}\left[\|\nabla F(x^k)-\overline{\nabla}f(x^k,y^{k+1})\|^2\right]+\alpha\delta T\mathbb{E}\left[\|\overline{\nabla}f(x^k,y^{k+1})-\frac{1}{T}\sum_{t=0}^{T-1}\bar{h}_f^{k,t}\|^2\right]\\ &\overset{(e)}{\le} \mathbb{E}[F(x^k)]-\frac{\alpha\delta T}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2]-\left(\frac{\alpha\delta T}{2}-\frac{\alpha^2\delta^2L_FT^2}{2}\right)\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]+\frac{\alpha^2\delta^2L_FT\tilde{\sigma}_f^2}{2}\\ &\quad+\alpha T\delta L_y^2\mathbb{E}\left[\|y^{k+1}-y^*(x^k)\|^2\right]+\mathcal{O}(\alpha^3T^3\delta)\end{aligned}$$

where (a) comes from the smoothness of F and the update (76), (b) is derived from (76), and (c) results from (84) and (85), (d) comes from $2a^\top b=\|a\|^2+\|b\|^2-\|a-b\|^2$ and $\|I-B^\dagger B\|\le 1$, (e) is derived from Lipschitz continuity of $\overline{\nabla}f(x,y)$, $F(x)=\overline{\nabla}f(x,y^*(x))$ and Lemma 25. This completes the proof. □

D.2 Error of lower-level upate

Lemma 27 (Lipschitz continuity and smoothness of the $r^*(x)$). *The projection offset $r^*(x)$ is L_r-Lipschitz continuous and L_{rx}-smooth with constants, respectively, defined as*

$$L_r:=\ell_{g,1}\left(1+L_y\right),\qquad L_{rx}:=\ell_{g,2}(1+L_y)^2+\ell_{g,1}L_{yx}.$$

Proof. Recall the definition of $r^*(x) := \nabla_y g(x, y^*(x))$, then for any x_1, x_2,

$$\begin{aligned}\|r^*(x_1) - r^*(x_2)\| &\le \|\nabla_y g(x_1, y^*(x_1)) - \nabla_y g(x_2, y^*(x_2))\| \\ &\le \ell_{g,1}\left(\|x_1 - x_2\| + \|y^*(x_1) - y^*(x_2)\|\right) \\ &\le \ell_{g,1}\left(1 + L_y\right)\|x_1 - x_2\| = L_r\|x_1 - x_2\|.\end{aligned}$$

Using the chain rule, we can obtain the gradient of $r^*(x)$ as

$$\nabla r^*(x) = \nabla_{yx} g(x, y^*(x)) + \nabla_{yy} g(x, y^*(x))\nabla y^*(x).$$

According to the Lipschitz continuity of $\nabla y^*(x)$ and $\nabla^2 g$, we get for any x_1, x_2

$$\begin{aligned}\|\nabla r^*(x_1) - \nabla r^*(x_2)\| \le{}& \|\nabla_{yx} g(x_1, y^*(x_1)) - \nabla_{yx} g(x_1, y^*(x_1))\| \\ &+ \|\nabla_{yy} g(x_1, y^*(x_1))\nabla y^*(x_1) - \nabla_{yy} g(x_2, y^*(x_2))\nabla y^*(x_2)\| \\ \le{}& \ell_{g,2}(1 + L_y)\|x_1 - x_2\| + \|\nabla_{yy} g(x_1, y^*(x_1))\|\|\nabla y^*(x_1) - \nabla y^*(x_2)\| \\ &+ \|\nabla y^*(x_2)\|\|\nabla_{yy} g(x_1, y^*(x_1)) - \nabla_{yy} g(x_2, y^*(x_2))\| \\ \le{}& \left(\ell_{g,2}(1 + L_y)^2 + \ell_{g,1} L_{yx}\right)\|x_1 - x_2\| = L_{rx}\|x_1 - x_2\|\end{aligned}$$

from which the proof is complete. □

Lemma 28 (Error of lower-level upate). *Suppose that Assumption 1–4 hold and $\beta \le \frac{1}{\ell_{g,1}}$, then the error of lower-level update can be bounded by*

$$\mathbb{E}\left[\|y^{k+1} - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^{k+1} - r^*(x^k)\|^2\right] \le (1-\nu)^S \,\mathbb{E}\left[\|y^k - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^k - r^*(x^k)\|^2\right] + S\beta^2\sigma_{g,1}^2 \tag{86a}$$

$$\begin{aligned}\mathbb{E}[\|y^{k+1} - y^*(x^{k+1})\|^2] \le{}& \left(1 + \gamma T + 2\eta L_{yx}\widetilde{C}_f^2 T^2\alpha^2\right)\mathbb{E}[\|y^{k+1} - y^*(x^k)\|^2] \\ &+ \left(L_y^2 + \frac{L_{yx}}{\eta}\right)\alpha^2 T\tilde{\sigma}_f^2 + \left(L_y^2 + \frac{L_{yx}}{\eta} + \frac{L_y^2}{\gamma}\right)\alpha^2 T^2\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]\end{aligned} \tag{86b}$$

$$\begin{aligned}\mathbb{E}[\|r^{k+1} - r^*(x^{k+1})\|^2] \le{}& \left(1 + \gamma T + 2\eta L_{rx}\widetilde{C}_f^2 T^2\alpha^2\right)\mathbb{E}[\|r^{k+1} - r^*(x^k)\|^2] \\ &+ \left(L_r^2 + \frac{L_{rx}}{\eta}\right)\alpha^2 T\tilde{\sigma}_f^2 + \left(L_r^2 + \frac{L_{rx}}{\eta} + \frac{L_r^2}{\gamma}\right)\alpha^2 T^2\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]\end{aligned} \tag{86c}$$

where $\widetilde{C}_f^2$ is defined in Lemma 21, $\nu := \min\left\{\beta\mu_g, p^2\right\}$, γ and η are the balancing constants will be chosen in the final theorem.

Proof. First, for a given x^k, defining $\nu := \min\left(\beta\mu_g, p^2\right)$ and applying Lemma C.1 and Lemma C.2 in (Mishchenko et al., 2022), we can obtain that

$$\begin{aligned}&\mathbb{E}\left[\|y^{k,s+1} - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^{k,s+1} - r^*(x^k)\|^2 \mid \mathcal{F}_{k,s}\right] \\ \le{}&(1-\nu)\left[\|y^{k,s} - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^{k,s} - r^*(x^k)\|^2\right] + \beta^2\sigma_{g,1}^2.\end{aligned} \tag{87}$$

Then taking expectation of the both sides of (87) and telescoping it, we can arrive at (86a).

Next, proof of (86b) and (86c) are similar with only difference on Lipschitz constant, so that we only prove (86c). For (86c), we have

$$\begin{aligned}\|r^{k+1} - r^*(x^{k+1})\|^2 = \|r^{k+1} - r^*(x^k)\|^2 &+ \underbrace{\|r^*(x^k) - r^*(x^{k+1})\|^2}_{J_1} \\ &+ 2\underbrace{\langle r^{k+1} - r^*(x^k), r^*(x^k) - r^*(x^{k+1})\rangle}_{J_2}.\end{aligned}$$

Since $r^*(x)$ is L_r Lipschitz continuous according to Lemma 27, J_1 can be bounded by

$$\begin{aligned}\mathbb{E}\left[J_1\right] &\leq L_r^2\mathbb{E}\left[\|x^{k+1}-x^k\|^2\right]\\ &\overset{(a)}{=} \alpha^2\delta^2L_r^2\mathbb{E}\left[\|\sum_{t=0}^{T-1}h_f^k\|_{P_x}^2\right]\\ &\overset{(b)}{\leq} \alpha^2\delta^2L_r^2\left(T^2\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]+T\tilde{\sigma}_f^2\right)\end{aligned} \tag{88}$$

where (a) comes from (58), (b) is attained by (84).

On the other hand, we can decompose J_2 by two terms as follows.

$$\begin{aligned}J_2 = &\underbrace{-\langle r^{k+1}-r^*(x^k),\nabla r^*(x^k)^\top(x^{k+1}-x^k)\rangle}_{J_{2,1}}\\ &\underbrace{-\langle r^{k+1}-r^*(x^k),r^*(x^{k+1})-r^*(x^k)-\nabla r^*(x^k)^\top(x^{k+1}-x^k)\rangle}_{J_{2,2}}.\end{aligned}$$

Moreover, the conditional expectation of $J_{2,1}$ can be bounded by

$$\begin{aligned}\mathbb{E}[J_{2,1}] &= -\mathbb{E}[\langle r^{k+1}-r^*(x^k),\nabla r^*(x^k)^\top(x^{k+1}-x^k)]\rangle\\ &= -\delta\mathbb{E}[\langle r^{k+1}-r^*(x^k),\nabla r^*(x^k)^\top(I-B^\dagger B)(\sum_{t=0}^{T-1}h_f^{k,t})\rangle]\\ &= -\delta\mathbb{E}\left[\sum_{t=0}^{T-1}\mathbb{E}[\langle r^{k+1}-r^*(x^k),\nabla r^*(x^k)^\top(I-B^\dagger B)h_f^{k,t}\rangle|\widetilde{\mathcal{F}}_k^t]\right]\\ &\leq -\alpha\delta T\mathbb{E}[\langle r^{k+1}-r^*(x^k),\nabla r^*(x^k)^\top(I-B^\dagger B)\bar{\bar{h}}_f^k\rangle]\\ &\overset{(a)}{\leq} \frac{\gamma}{2}\mathbb{E}[\|r^{k+1}-r^*(x^k)\|^2]+\frac{\alpha^2\delta^2T^2L_r^2}{2\gamma}\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]\end{aligned} \tag{89}$$

Based on the smoothness of $r^*(x)$ in Lemma 27 and Young's inequality, $J_{2,2}$ can be bounded by

$$\begin{aligned}\mathbb{E}[J_{2,2}] &\leq \mathbb{E}\left[\|r^{k+1}-r^*(x^k)\|\|r^*(x^{k+1})-r^*(x^k)-\nabla r^*(x^k)^\top(x^{k+1}-x^k)\|^2\right]\\ &\leq \frac{L_{rx}}{2}\mathbb{E}\left[\|r^{k+1}-r^*(x^k)\|\|x^{k+1}-x^k\|^2\right]\\ &\leq \frac{\eta L_{rx}\alpha^2\delta^2}{2}\mathbb{E}[\|r^{k+1}-r^*(x^k)\|^2\|\sum_{t=0}^{T-1}h_f^{k,t}\|_{P_x}^2]+\frac{L_{rx}\alpha^2\delta^2}{2\eta}\mathbb{E}\left[\|\sum_{t=0}^{T-1}h_f^{k,t}\|_{P_x}^2\right]\\ &\leq \frac{\eta L_{rx}\alpha^2\delta^2T}{2}\mathbb{E}\left[\|r^{k+1}-r^*(x^k)\|^2\sum_{t=0}^{T-1}\mathbb{E}[\|h_f^{k,t}\|^2|\widetilde{\mathcal{F}}_k^t]\right]+\frac{L_{rx}\alpha^2\delta^2T^2}{2\eta}\left(\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]+\frac{\tilde{\sigma}_f^2}{T}\right)\\ &\leq \eta L_{rx}\alpha^2\delta^2T\mathbb{E}\left[\|r^{k+1}-r^*(x^k)\|^2\sum_{t=0}^{T-1}(\mathbb{E}[\|\bar{h}_f^{k,t}\|^2|\widetilde{\mathcal{F}}_k^t]+\tilde{\sigma}_f^2)\right]+\frac{L_{rx}\alpha^2\delta^2T^2}{2\eta}\left(\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]+\frac{\tilde{\sigma}_f^2}{T}\right)\\ &\overset{(a)}{\leq} \eta L_{rx}\alpha^2\delta^2\tilde{C}_f^2T^2\mathbb{E}\left[\|r^{k+1}-r^*(x^k)\|^2\right]+\frac{L_{rx}\alpha^2\delta^2T^2}{2\eta}\left(\mathbb{E}[\|\bar{\bar{h}}_f^k\|_{P_x}^2]+\frac{\tilde{\sigma}_f^2}{T}\right)\end{aligned} \tag{90}$$

where (a) comes from

$$\begin{aligned}\mathbb{E}[\|\bar{h}_f^{k,t}\|^2|\widetilde{\mathcal{F}}_k^t] &\leq 2\mathbb{E}[\|\bar{h}_f^{k,t}-G(x^{k,t},y^{k+1},x^k)\|^2|\widetilde{\mathcal{F}}_k^t]+2\mathbb{E}[\|G(x^{k,t},y^{k+1},x^k)\|^2|\widetilde{\mathcal{F}}_k^t]\\ &\leq 2b_k^2+2\ell_{f,0}^2(1+L_y)^2\end{aligned}$$

and $\tilde{C}_f^2=2b_k^2+2\ell_{f,0}^2(1+L_y)^2+\tilde{\sigma}_f^2$.

Then combining (88), (89) and (90), we get

$$\mathbb{E}[\|r^{k+1}-r^*(x^{k+1})\|^2] \le \left(1+\gamma+2\eta L_{rx}\widetilde{C}_f^2 T^2\alpha^2\delta^2\right)\mathbb{E}[\|r^{k+1}-r^*(x^k)\|^2] + \left(L_r^2+\frac{L_{rx}}{\eta}\right)\alpha^2\delta^2 T\tilde{\sigma}_f^2 + \left(L_r^2+\frac{L_{rx}}{\eta}+\frac{L_r^2}{\gamma}\right)\alpha^2\delta^2T^2\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right].$$

which completes the proof for (86b). With similar proof for (86c), we can prove (86b). □

D.3 Proof of Theorem 5

We first restate the Theorem 5 in a formal way as follows.

Restatement of Theorem 5. *Under Assumption 1–4, defining the constants as*

$$\bar{\alpha}_1 = \frac{1}{2L_F + 4L_fL_r + \frac{4L_fL_{rx}}{L_r} + \frac{\left(5L_fL_y+\eta L_{yx}\tilde{C}_f^2\right)\left(1+4L_fL_r+\frac{4L_fL_{rx}}{\eta L_r}\right)}{\mu_g^2}},$$
$$\bar{\alpha}_2 = \frac{\mu_g}{\ell_{g,1}(5L_fL_r+\eta L_{rx}\tilde{C}_f^2)}$$

and choosing

$$\alpha = \min\left(\bar{\alpha}_1, \bar{\alpha}_2, \frac{\bar{\alpha}}{\sqrt{K}}\right), \qquad \beta = \frac{5L_fL_r+\eta L_{rx}\tilde{C}_f^2}{\mu_g}\alpha, \qquad N = \mathcal{O}(\log K)$$

then for any $S \ge 1$, we have

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] = \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{K}}\right).$$

Proof. First, without loose of generality, we can assume that $\ell_{g,1} \ge 1$ so that $L_r \ge L_y, L_{rx} \ge L_{yx}$ and plugging (86b), (86c) into (86a) in Lemma 28, we get that

$$\begin{aligned}
&\mathbb{E}\left[\|y^{k+1}-y^*(x^{k+1})+\frac{\beta^2}{p^2}\|r^{k+1}-r^*(x^{k+1})\|^2\|^2\right] \\
&\le \left(1+\gamma+2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2T^2\right)(1-\nu)^S\,\mathbb{E}\left[\|y^k-y^*(x^k)+\frac{\beta^2}{p^2}\|r^k-r^*(x^k)\|^2\|^2\right] \\
&\quad + \left(1+\gamma+2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2T^2\right)S\beta^2\sigma_{g,1}^2 + \left(L_r^2+\frac{L_{rx}}{\eta}\right)\left(1+\frac{\beta^2}{p^2}\right)\alpha^2\delta^2T\tilde{\sigma}_f^2 \\
&\quad + \left(L_r^2+\frac{L_{rx}}{\eta}+\frac{L_r^2}{\gamma}\right)\left(1+\frac{\beta^2}{p^2}\right)\alpha^2\delta^2T^2\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right].
\end{aligned} \tag{91}$$

Then using Lyapunov function defined in (13) and applying (91), Lemma 26 and Lemma 28, we get

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{V}^{k+1}\right]-\mathbb{E}\left[\mathbb{V}^k\right] \\
&\le -\frac{\alpha\delta T}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \alpha\delta TL_f^2\mathbb{E}\left[\|y^{k+1}-y^*(x^k)\|^2\right] + \mathcal{O}(\alpha^3T^3\delta) \\
&\quad - \left(\frac{\alpha\delta T}{2}-\frac{L_F\alpha^2\delta^2T^2}{2}\right)\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right] + \frac{L_F\alpha^2\delta^2T\tilde{\sigma}_f^2}{2} \\
&\quad + \frac{L_f}{L_r}\left[\left(1+\gamma+2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2T^2\right)(1-\nu)^S-1\right]\mathbb{E}\left[\|y^k-y^*(x^k)\|^2+\frac{\beta^2}{p^2}\|r^k-r^*(x^k)\|^2\right] \\
&\quad + \frac{L_f}{L_r}\left(1+\gamma+2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2T^2\right)S\beta^2\sigma_{g,1}^2 + \frac{L_f}{L_r}\left(L_r^2+\frac{L_{rx}}{\eta}\right)\left(1+\frac{\beta^2}{p^2}\right)\alpha^2\delta^2T\tilde{\sigma}_f^2 \\
&\quad + \frac{L_f}{L_r}\left(L_r^2+\frac{L_{rx}}{\eta}+\frac{L_r^2}{\gamma}\right)\left(1+\frac{\beta^2}{p^2}\right)\alpha^2\delta^2T^2\mathbb{E}\left[\|\bar{\bar{h}}_f^k\|_{P_x}^2\right]
\end{aligned}$$

$$
\begin{aligned}
&\leq -\frac{\alpha\delta T}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \frac{L_f}{L_r}\left(1+\gamma+L_rL_f\alpha\delta T + 2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2T^2\right)\beta^2 S\sigma_{g,1}^2 + \mathcal{O}(\alpha^3T^3\delta)\\
&\quad + \left[\frac{L_F}{2} + \frac{L_f}{L_r}\left(L_r^2 + \frac{L_{rx}}{\eta}\right)\left(1+\frac{\beta^2}{p^2}\right)\right]\alpha^2T\delta^2\tilde{\sigma}_f^2\\
&\quad - \left[\frac{\alpha\delta T}{2} - \left(\frac{L_F}{2} + L_fL_r\left(1+\frac{1}{\gamma}\right)\left(1+\frac{\beta^2}{p^2}\right) + \frac{L_fL_{rx}}{\eta L_r}\left(1+\frac{\beta^2}{p^2}\right)\right)\alpha^2\delta^2T^2\right]\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right]\\
&\quad - \left(\frac{L_f\nu}{L_r} - \alpha\delta TL_f^2 - \frac{L_f\gamma}{L_r} - \frac{2\eta L_fL_{rx}\tilde{C}_f^2\alpha^2\delta^2T^2}{L_r}\right)\mathbb{E}\left[\|y^k - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^k - r^*(x^k)\|^2\right]. \qquad (92)
\end{aligned}
$$

Selecting $\gamma = 4L_fL_r\alpha\delta T, p = \sqrt{\beta\mu_g}$, (92) can be simplified by

$$
\begin{aligned}
&\mathbb{E}[\mathbb{V}^{k+1}] - \mathbb{E}[\mathbb{V}^k]\\
&\leq -\frac{\alpha\delta T}{2}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] + \frac{L_f}{L_r}\left(1+5L_fL_r\alpha\delta T + 2\eta L_{rx}\widetilde{C}_f^2\alpha^2\delta^2\right)\beta^2 S\sigma_{g,1}^2 + \mathcal{O}(\alpha^3T^3\delta)\\
&\quad + \left[\frac{L_F}{2} + \frac{L_f}{L_r}\left(L_r^2 + \frac{L_{rx}}{\eta}\right)\left(1+\frac{\beta}{\mu_g}\right)\right]\alpha^2\delta^2T\tilde{\sigma}_f^2\\
&\quad - \left[\frac{\alpha\delta T}{4} - \left(\frac{L_F}{2} + L_fL_r + \frac{L_fL_{rx}}{\eta L_r}\right)\alpha^2\delta^2T^2 - \frac{\beta\alpha\delta T}{4\mu_g} - \left(L_fL_r + \frac{L_fL_{rx}}{\eta L_r}\right)\frac{\beta\alpha^2\delta^2T^2}{\mu_g}\right]\mathbb{E}\left[\|\bar{h}_f^k\|_{P_x}^2\right]\\
&\quad - \left(\frac{L_f\mu_g\beta}{L_r} - 5\alpha\delta L_f^2T - \frac{2\eta L_fL_{rx}\tilde{C}_f^2\alpha^2\delta^2T^2}{L_r}\right)\mathbb{E}\left[\|y^k - y^*(x^k)\|^2 + \frac{\beta^2}{p^2}\|r^k - r^*(x^k)\|^2\right]. \qquad (93)
\end{aligned}
$$

Let $\alpha\delta T \leq 1$ and since we also need $\beta \leq \frac{1}{\ell_{g,1}}$, then the sufficient condition of making the last two terms negative becomes

$$
\alpha\delta \leq \frac{\bar{\alpha}}{T}\,, \beta = \frac{5L_fL_r + 2\eta L_{rx}\tilde{C}_f^2}{\mu_g}\alpha\delta T
$$

where

$$
\begin{aligned}
\bar{\alpha}_1 &= \frac{1}{2L_F + 4L_fL_r + \frac{4L_fL_{rx}}{\eta L_r} + \frac{\left(5L_fL_y + \eta L_{yx}\tilde{C}_f^2\right)\left(1+4L_fL_r + \frac{4L_fL_{rx}}{\eta L_r}\right)}{\mu_g^2}},\\
\bar{\alpha}_2 &= \frac{\mu_g}{\ell_{g,1}(5L_fL_r + \eta L_{rx}\tilde{C}_f^2)},\ \ \bar{\alpha} = \min\left(\bar{\alpha}_1, \bar{\alpha}_2\right). \qquad (94)
\end{aligned}
$$

Then (93) becomes

$$
\frac{\alpha\delta T}{2}\mathbb{E}[\|\nabla F(x^k)\|_{P_x}^2] \leq \left(\mathbb{E}[\mathbb{V}^k] - \mathbb{E}[\mathbb{V}^{k+1}]\right) + c_1S\alpha^2\delta^2T^2\sigma_{g,1}^2 + c_2\alpha^2\delta^2T\tilde{\sigma}_f^2 + \mathcal{O}(\alpha^3\delta^3T^2) \qquad (95)
$$

where c_1 and c_2 are defined as

$$
\begin{aligned}
c_1 &= \frac{L_fS}{L_r}\left(1+\eta L_{rx}\widetilde{C}_f^2\bar{\alpha}^2\right)\left(\frac{5L_fL_r + 2\eta L_{rx}\tilde{C}_f^2}{\mu_g}\right)^2\\
c_2 &= \frac{L_F}{2} + \frac{L_f}{L_r}\left(L_r^2 + \frac{L_{rx}}{\eta}\right) \qquad (96)
\end{aligned}
$$

Telescoping (95) and dividing both sides by $\alpha\delta TK$ leads to

$$
\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla F(x^k)\|_{P_x}^2\right] \leq \frac{2(\mathbb{V}^0 - F^*)}{\alpha\delta TK} + 2c_1\sigma_{g,1}^2\alpha\delta T + 2c_2\tilde{\sigma}_f^2\alpha\delta + \mathcal{O}(\alpha^2\delta^2T).
$$

This completes the proof. □

E Application on federated bilevel learning

In this section, we present the pseudo-code of E-AiPOD on federated bilevel learning and some omitted derivations.

E.1 Pseudo-code of Algorithm 2 on federated bilevel learning

In federated bilevel setting, if we plug V_2 in (19) to (9a), then we obtain $w^k = [w_1^k, \cdots, w_M^k]$ with

$$w_m^k = -\nabla_{xy} g(x_m^k, y_m^{k+1}; \phi_{(0)}^k) \left[\frac{\tilde{c}N}{\ell_{g,1}} \prod_{n=1}^{N'} \left(I - \frac{\tilde{c}}{M\ell_{g,1}} \sum_{i=1}^{M} \nabla_{yy} g(x_i^k, y_i^{k+1}; \phi_{(n)}^k) \right) \right] \left(\frac{1}{M} \sum_{i=1}^{M} \nabla_y f(x_i^k, y_i^{k+1}; \xi_i^k) \right). \tag{97}$$

To avoid transmission of Hessian, which is highly expensive, we can calculate w_m^k by communicating Hessian-vector product purely. We detail the full procedure in Algorithm 4. Note that the output of Algorithm 4 is exactly w_m^k. We set the number of LL loop $S = 1$ and summarize the E-AiPOD on federated bilevel learning in Algorithm 3.

Algorithm 3 E-AiPOD in federated bilevel learning: blue part denotes the LL update; red part is the UL update

Initialization: $\{x_m^0, y_m^0\}_{m\in[M]}$, stepsizes $\{\alpha, \beta, \delta\}$, projection probability p, N.
for $k = 0$ **to** $K - 1$ **do**
  for all workers $m \in [M]$ in parallel **do** ▷ LL update
    update $\hat{y}_m^{k+1} = y_m^k - \beta(\nabla_y g_m(x_m^k, y_m^k; \phi_m^k) - r_m^k)$
    draw a Bernoulli $\theta^k \in \{0, 1\}$ with probability p
    if $\theta^{k,s} = 1$ **then**
      update $y_m^{k+1} = \frac{1}{M} \sum_{i=1}^{M} \left(\hat{y}_i^{k+1} - \frac{\beta}{p} r_i^k \right)$ ▷ Communicate to server and average
      update $r_m^{k+1} = r_m^k + \frac{p}{\beta}(y_m^{k+1} - \hat{y}_m^{k+1})$
    else
      set $y_m^{k+1} = \hat{y}_m^{k+1}, \quad r_m^{k+1} = r_m^k$
    end if
  end for
  for all workers $m \in [M]$ in parallel **do** ▷ UL update
    calculate w_m^k by (97) ▷ Call Algorithm 4
    for $t = 0$ **to** $T - 1$ **do**
      update $x_m^{k,t+1} = x_m^{k,t} - \alpha(\nabla_x f_m(x_m^k, y_m^{k+1}; \xi_m^{k,t}) + w_m^k)$ ▷ Initialize $x_m^{k,0} = x_m^k$
    end for
    set $\Delta_m^k = x_m^{k,T} - x_m^k$
  end for
  update $x_m^{k+1} = x_m^k + \delta \sum_{m=1}^{M} \Delta_m^k$ ▷ Communicate to server
end for

E.2 Equivalence between our metric with metric in federated bilevel learning

In this section, we prove the equivalence of our measure in (4) with the measure of FedNest in federated bilevel setting.

According to Lemma 12, and $\mathcal{X} = \{x \mid Bx = 0\}$ in federated bilevel setting, we know

$$(I - B^\dagger B)\nabla F(x) = \mathrm{Proj}_{\mathcal{X}}(\nabla F(x)). \tag{98}$$

Then according to (21a) and (20a), we obtain

$$\begin{aligned}(I - B^\dagger B)\nabla F(x) =& \frac{1}{M} \sum_{m=1}^{M} \nabla_{x_m} f_m(x_m, y_m^*(x_m)) + \left(\frac{1}{M} \sum_{m=1}^{M} \nabla_{x_m y_m} g_m(x_m, y_m^*(x_m)) \right) \\ & \times \left(\frac{1}{M} \sum_{m=1}^{M} \nabla_{y_m y_m} g_m(x_m, y_m^*(x_m)) \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} \nabla_{y_m} f_m(x_m, y_m^*(x_m)) \right). \end{aligned} \tag{99}$$

Algorithm 4 Efficient calculation of $\{w_m^k\}_{m\in[M]}$: green part denotes the communication round

Initialization: $\{x_m^k, y_m^{k+1}\}_{m\in[M]}$, constant $\tilde{c} \leq 1, \ell_{g,1}, N$.
Draw $N' \in \{0, \cdots N-1\}$ uniformly
for all workers $m \in [M]$ in parallel **do**
  update $v_{m,0}^k = \nabla_y f_m(x_m^k, y_m^{k+1}; \phi_{m,0}^k)$
end for
update $v_0^k = \frac{\tilde{c}N}{\ell_{g,1}M} \sum_{m=1}^M v_{m,0}^k$ ▷ Communicate to server
for $n = 1$ **to** N' **do**
  for all workers $m \in [M]$ in parallel **do**
    update $v_{m,n}^k = (I - \frac{\tilde{c}}{\ell_{g,1}} \nabla_{yy} g_m(x_m^k, y_m^{k+1}; \phi_{m,n}^k)) v_{n-1}^k$
  end for
  $v_n^k = \frac{1}{M} \sum_{m=1}^M v_{m,n}^k$ ▷ Communicate to server
end for ▷ Set $v^k = v_{N'}^k$
for all workers $m \in [M]$ in parallel **do**
  $w_m^k = -\nabla_{xy} g(x_m^k, y_m^k; \phi_{(0)}^k) v^k$.
end for

On the other hand, the gradient of the objective in FedNest is

$$\frac{1}{M}\sum_{m=1}^M \nabla_x f_m(x, y^*(x)) + \left(\frac{1}{M}\sum_{m=1}^M \nabla_{xy} g_m(x, y^*(x))\right) \tag{100}$$

$$\times \left(\frac{1}{M}\sum_{m=1}^M \nabla_{yy} g_m(x, y^*(x))\right)^{-1} \left(\frac{1}{M}\sum_{m=1}^M \nabla_{y_m} f_m(x, y^*(x))\right). \tag{101}$$

We find that (101) is the same as (99), if replacing $g_m(x, y^*(x)), f_m(x, y^*(x))$ by $g_m(x_m, y_m^*(x_m)), f_m(x_m, y_m^*(x_m))$. Moreover, since $\mathbb{E}[\|\nabla F(x)\|_{P_x}^2] = \mathbb{E}[\|(I - B^\dagger B)\nabla F(x)\|^2]$, the measure $\mathbb{E}[\|\nabla F(x)\|_{P_x}^2]$ in our analysis coincides with the gradient norm measure in FedNest.

E.3 Additional related works on federated bilevel learning

Federated learning and the Federated average (FedAvg) algorithm were first introduced by (McMahan et al., 2017). The convergence rate of FedAvg has has been thoroughly investigated by (Stich, 2019; Stich and Karimireddy, 2020; Yu et al., 2019; Woodworth et al., 2020; Yang et al., 2021a); see a survey (Kairouz et al., 2021). Later on, (Mitra et al., 2021) applied variance reduction techniques to tackle the heterogeneous data and obtain the linear convergence for strongly convex objectives. Very recently, Mishchenko et al. (2022) has first theoretically achieved the optimal complexity for strongly convex objectives without assuming any similarity. Meanwhile, Tarzanagh et al. (2022) first proposed the federated bilevel framework owing to the vast potential applications with the nested structure and achieved $\tilde{\mathcal{O}}(\epsilon^{-2})$ rate of both sample complexity and communication complexity. Later on, Li et al. (2022b) enhanced the convergence rate to $\tilde{\mathcal{O}}(\epsilon^{-1.5})$ by momentum-based variance-reduction technique but it required additional transmission of momentum parameters and bounded data similarity assumption. Huang et al. (2022) introduced a finite-sum framework to learn adaptively weighted node in federated learning via deterministic bilevel optimization. Recent advances on decentralized bilevel optimization (Lu et al., 2022; Yang et al., 2022; Gao et al., 2022a) and (Chen et al., 2022b) focus on different settings. While it is not the focus here, it would be interesting to extend E-AiPOD to the decentralized settings in future work.

E.4 Comparison with the state-of-the-art work on federated bilevel learning

In this section, we compare the theoretical results of our methods in federated bilevel learning settings over the state-of-the-art works in Table 2. Note that the communication complexity of FedBiOAcc in (Li et al., 2022b) inherits from its non-periodic counterpart SUSTAIN (Khanduri et al., 2021), whose sample complexity is $\tilde{\mathcal{O}}(\epsilon^{-1.5})$ owing to the momentum.

	AiPOD	E-AiPOD	FedNest	FedBiOAcc
y-update	AvgSGD	Scaffnew	variance reduction	momentum
x-update	AvgSGD	FedAvg	variance reduction	momentum
communicated parameters	x, y	x, y	x, y	x, y and momentums
UL communication	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-2}/T)$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-1.5})$
LL communication	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-1.5}/T^{\frac{3}{4}})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-1.5})$

Table 2: Communication comparison of AiPOD, E-AiPOD in our paper and the state-of-the-art works on stochastic federated bilevel learning (FedNest in (Tarzanagh et al., 2022), FedBioAcc in (Li et al., 2022b)) to achieve ϵ stationary point. AvgSGD means FedAvg with averaging period 1 and Scaffnew denotes (Mishchenko et al., 2022).

F Additional Details of Experiments

In this section, we will report the detailed settings of the experiments in Section 5. In the federated bilevel learning experiments, the number of workers is set as $M = 50$ and each local network is a 2-layer multilayer perceptron with hidden dimension 200. The hyper-parameters are found by measuring both the convergence speed and the stability of the algorithm via a grid search, and we report them here.

F.1 Synthetic task

E-AiPOD: The projection probability is set as $p = 0.3$ in the right figures, the total number of iterations is $K = 400/p$, the number of UL iterations is $T = 2$ in the left figures, the number of LL iterations is $S = 5$, the step sizes are set as $\alpha = 0.02, \beta = 0.01$, the noise has mean 0 and std 0.1. AiPOD is a special case of E-AiPOD with $p = 1$.

F.2 Federated representation learning

In this section, we apply E-AiPOD in Algorithm 2 to the federated representation learning task. The classic machine learning approach learns a data representation and a downstream header jointly on the training data set. While the bilevel representation learning (Franceschi et al., 2018) seeks to learn a data representation on the validation set and a header on the training data set, the procedure can then be formulated as a bilevel problem. In a federated representation learning setting with $M = 50$ clients, the validation and training data sets are distributed among clients, and the goal is to learn a representation and header respectively on the joint validation and training data set while protecting data privacy. Formally, the problem can be formulated as a case of (18), given by

$$\begin{aligned}
&\min_{x\in\mathcal{X}} \frac{1}{M}\sum_{m=1}^{M} f_{\text{ce}}(x_m, y_m^*(x); \mathcal{D}_{\text{val}}^m),\\
&\text{s.t. } y^*(x) = \arg\min_{y\in\mathcal{Y}} \frac{1}{M}\sum_{m=1}^{M} f_{\text{ce}}(x, y_m; \mathcal{D}_{\text{tr}}^m) + 0.05\|y_m\|^2,
\end{aligned}$$

where x is the parameters of the representation layer; y is the parameter of the classifier layer; $\mathcal{D}_{\text{tr}}^m$ and $\mathcal{D}_{\text{val}}^m$ are, respectively, the training and validation set of client m; $\mathcal{X}$ and $\mathcal{Y}$ are the consensus sets defined in (18). The cross-entropy loss f_{ce} is defined as

$$f_{\text{ce}}(x, y; \mathcal{D}) := -\frac{1}{|\mathcal{D}|}\sum_{d_n\in\mathcal{D}} \log \frac{\exp\left(h_{l_n}(x, y; d_n)\right)}{\sum_{c=1}^{C}\exp\left(h_c(x, y; d_n)\right)}$$

where C is the number of classes, d_n is the n-th data from class l_n in data set $\mathcal{D}$ and $h(x, y; d_n) = [h_1(x, y; d_n), ..., h_C(x, y; d_n)]^\top \in \mathbb{R}^C$ is the output of the model with parameter (x, y) and input d_n.

Hyperparameters. E-AiPOD: The communication probability is set as $p = 0.1$ in Figure 2 (right), $S = 20$, $\alpha = 0.01, \beta = 0.05$, Neumann iteraion $N' = 5$, and the batch size is 256. FedNest (notations in Tarzanagh et al. (2022)): Choose LL iteration number $\tau = 10$ and episode $T = 1$ so that the communication frequency is 0.1 per LL iteration, which is the same as the choice of p for E-AiPOD. The UL iteration numbers are specified in Figure 2, and we set $\alpha = 0.01$, $\beta = 0.02$ (under $T = 1$) or $\beta = 0.01$ (under $T = 5$), $N' = 5$ and batch size as 256.

F.3 Federated loss function tuning

In this subsection, we apply E-AiPOD to the federated learning from imbalanced data task, where the goal is to learn a good model that guarantees both the fairness and generalization from datasets with under-represented classes (Li et al., 2021). In the UL, the loss-tuning parameters are trained to improve generalization and fairness, while the model parameters are trained on a possibly imbalanced data-set in the LL. The method was later extended to the federated setting in (Tarzanagh et al., 2022). Formally, the problem can be written as a case of (18), given by

$$\min_{x\in\mathcal{X}} \frac{1}{M}\sum_{m=1}^{M} f_{\rm vs}^{\rm up}(y_m^*(x);\mathcal{D}_{\rm val}^m),$$
$$\text{s.t. } y^*(x) = \arg\min_{y\in\mathcal{Y}} \frac{1}{M}\sum_{m=1}^{M} f_{\rm vs}^{\rm low}(x,y_m;\mathcal{D}_{\rm tr}^m),$$

where the number of clients is $M = 50$, x is the loss-tuning parameters and y is the parameter of the neural network. Here $\mathcal{D}_{\rm tr}^m$ and $\mathcal{D}_{\rm val}^m$ are respectively the training and validation set of client m and $\mathcal{X}, \mathcal{Y}$ are the consensus sets defined in (18). The numbers of data of different classes are imbalanced in the training data-set $\{\mathcal{D}_{\rm tr}^m\}_{m=1}^M$. Introduced in (Kini et al., 2021), the so-called vector-scaling loss $f_{\rm vs}^{\rm low}$ is defined as

$$f_{\rm vs}^{\rm low}(x,y;\mathcal{D}) := -\frac{1}{|\mathcal{D}|}\sum_{d_n\in\mathcal{D}} \omega_{l_n} \log \frac{\exp\left(\delta_{l_n} h_{l_n}(y;d_n) + \tau_{l_n}\right)}{\sum_{c=1}^{C}\exp\left(\delta_c h_c(y;d_n) + \tau_c\right)}$$

where N is the data set size, C is the number of classes, d_n is the n-th data with label class l_n in data set $\mathcal{D}$ and $h(y;d_n) = [h_1(y;d_n),...,h_C(y;d_n)]^\top \in \mathbb{R}^C$ is the logit output of the neural network with parameter y and input d_n. Define $x = (\omega, \delta, \tau)$ where $\omega := [\omega_1,...,\omega_C]^\top \in \mathbb{R}^C$ and δ, τ can be defined similarly. The upper-level loss $f_{\rm vs}^{\rm up}$ is a special case of $f_{\rm vs}^{\rm low}$ with $\delta = 1$,$\tau = 0$ and ω is a fixed class weight vector for the validation data set.

Hyperparameters. E-AiPOD: Communication probability $p = 0.3$ in Figure 2 (right), $S = 20$, $\alpha = 0.01, \beta = 0.04$, $N' = 3$, and batch size 256. FedNest: Choose LL iteration number $\tau = 3$ and episode $T = 3$ and thus the communication frequency is 0.3 per LL iteration, which is the same as $p = 0.3$. The UL iteration numbers are specified in Figure 2. Set $\alpha = 0.01$, $\beta = 0.02$, $N' = 3$, and batch size as 256.