A Nearly Optimal Single Loop Algorithm for Stochastic Bilevel Optimization under Unbounded Smoothness

Xiaochuan Gong [1] **Jie Hao** [1] **Mingrui Liu** [1]

Abstract

This paper studies the problem of stochastic bilevel optimization where the upper-level function is nonconvex with potentially unbounded smoothness and the lower-level function is strongly convex. This problem is motivated by meta-learning applied to sequential data, such as text classification using recurrent neural networks, where the smoothness constant of the upper-level loss function scales linearly with the gradient norm and can be potentially unbounded. Existing algorithm crucially relies on the nested loop design, which requires significant tuning efforts and is not practical. In this paper, we address this issue by proposing a Single Loop bIlevel oPtimizer (SLIP). The proposed algorithm first updates the lower-level variable by a few steps of stochastic gradient descent, and then simultaneously updates the upper-level variable by normalized stochastic gradient descent with momentum and the lower-level variable by stochastic gradient descent. Under standard assumptions, we show that our algorithm finds an ϵ-stationary point within $\widetilde{O}(1/\epsilon^4)$[1] oracle calls of stochastic gradient or Hessian-vector product, both in expectation and with high probability. This complexity result is nearly optimal up to logarithmic factors without mean-square smoothness of the stochastic gradient oracle. Our proof relies on (i) a refined characterization and control of the lower-level variable and (ii) establishing a novel connection between bilevel optimization and stochastic optimization under distributional drift. Our experiments on various tasks show that our algorithm significantly outperforms strong baselines in bilevel optimization. The code is available here.

[1]Department of Computer Science, George Mason University, USA. Correspondence to: Mingrui Liu <mingruil@gmu.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

[1]Here $\widetilde{O}(\cdot)$ compresses logarithmic factors of $1/\epsilon$ and $1/\delta$, where $\delta \in (0, 1)$ denotes the failure probability.

1. Introduction

There has been a surge in interest in bilevel optimization, driven by its broad applications in machine learning. This includes meta-learning (Franceschi et al., 2018; Rajeswaran et al., 2019), hyperparameter optimization (Franceschi et al., 2018; Feurer & Hutter, 2019), fair model training (Roh et al., 2020), continual learning (Borsos et al., 2020; Hao et al., 2023), and reinforcement learning (Konda & Tsitsiklis, 1999). The bilevel optimization has the following form:

$$\min_{x\in\mathbb{R}^{d_x}} \Phi(x) := f(x, y^*(x)), \quad \text{s.t.,} \quad y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_y}} g(x, y), \tag{1}$$

where f and g are upper-level and lower-level functions respectively. For example, in meta-learning (Finn et al., 2017; Franceschi et al., 2018), the bilevel formulation is trying to find a common representation such that it can quickly adapt to different tasks by adjusting its task-specific head, where x denotes the shared representation across different tasks, y denotes the task-specific head. We consider the stochastic optimization setting, where $f(x, y) = \mathbb{E}_{\xi\sim\mathcal{D}_f}[F(x, y; \xi)]$ and $g(x, y) = \mathbb{E}_{\zeta\sim\mathcal{D}_g}[G(x, y; \zeta)]$, where $\mathcal{D}_f$ and $\mathcal{D}_g$ are underlying data distributions for f and g respectively.

Most theoretical studies and algorithmic developments on bilevel optimization typically assume that the upper-level function is smooth and nonconvex (i.e., the gradient is Lipschitz), and the lower-level function is strongly convex (Ghadimi & Wang, 2018; Hong et al., 2023; Grazzi et al., 2022; Ji et al., 2021). However, recent work shows that there is a class of neural networks whose smoothness parameter is potentially unbounded (Zhang et al., 2020c; Crawshaw et al., 2022), such as recurrent neural networks (RNNs) (Elman, 1990), long-short-term memory networks (LSTMs) (Hochreiter & Schmidhuber, 1997) and transformers (Vaswani et al., 2017). Therefore, it is important to design algorithms when the upper-level function has an unbounded smoothness parameter. Recently, (Hao et al., 2024) designed an algorithm under this regime and proved its convergence to stationary points. However, the structure of their algorithm is rather complex: it has a nested double loop to update the lower-level variable, which requires significant tuning efforts. It remains unclear how to design a provably efficient single-loop algorithm under this setting.

Designing a single-loop algorithm for bilevel problem with a unbounded smooth upper-level function has the following challenges. First, it is difficult to apply the similar analysis strategy of previous work (Hao et al., 2024) to any single-loop algorithms. Hao et al. (2024) designed a complicated procedure to update the lower-level variable to obtain an accurate estimator for the optimal lower-level solution at every iteration, but single-loop algorithms typically cannot achieve the same goal. Second, we need to simultaneously control the progress and their mutual dependence on the upper-level and lower-level problems, even if the lower-level variable may not be accurate.

To address these challenges, we propose a new single-loop bilevel optimizer, namely SLIP. The algorithm is surprisingly simple: after a few iterations of stochastic gradient descent (SGD) to update the lower-level variable, the algorithm simultaneously updates the upper-level variable by normalized stochastic gradient descent with momentum and updates the lower-level variable by SGD. It does not perform periodic updates of the lower-level variable as required in (Hao et al., 2024). The algorithm analysis relies on two important new techniques. First, we provide a refined characterization and control for the lower-level variable so that there is no need to get an accurate estimate of the lower-level variable at every iteration, while we can still ensure convergence. Second, we establish a novel connection between bilevel optimization and stochastic optimization under distributional drift (Cutler et al., 2023), and it helps us analyze the error of the lower-level variable. Our contributions of this paper are summarized as follows.

- We design a new single-loop algorithm named SLIP, for solving stochastic bilevel optimization problem where the upper-level function is nonconvex and unbounded smooth and the lower-level function is strongly convex. To the best of our knowledge, this is the first single-loop algorithm in this setting.
- We prove that the algorithm converges to the ϵ-stationary point within $\widetilde{O}(1/\epsilon^4)$ oracle calls of the stochastic gradient or the Hessian-vector product, both in expectation and with high probability. This complexity result is nearly optimal up to logarithmic factors without mean-square smoothness of the stochastic gradient oracle (A summary of our results and a comparison to prior work are provided in Table 1 at Appendix A). The proof relies on a novel characterization of the lower-level variable and a novel connection between bilevel optimization and stochastic optimization with distributional drift, which are new and not leveraged by previous work on bilevel optimization.
- We perform extensive experiments on hyper-representation learning and data hypercleaning on text classification tasks. Our algorithm shows significant speedup compared with strong baselines in bilevel optimization.

2. Related Work

Bilevel Optimization. Bilevel optimization was introduced by (Bracken & McGill, 1973). Some classical algorithms were proposed for certain classes of bilevel optimization and asymptotic convergence was provided (Vicente et al., 1994; Anandalingam & White, 1990; White & Anandalingam, 1993). Recently, Ghadimi & Wang (2018) pioneered the non-asymptotic convergence of gradient-based methods for bilevel optimization when the lower-level problem is strongly convex. This complexity result was improved by a series of work (Hong et al., 2023; Chen et al., 2021; 2022; Ji et al., 2021; Chen et al., 2023b). When the function has a mean squared smooth stochastic gradient, the complexity was further improved by incorporating variance reduction and momentum techniques (Khanduri et al., 2021; Dagréou et al., 2022; Guo et al., 2021; Yang et al., 2021). There is a line of work that designed fully first-order algorithms for stochastic bilevel optimization problems (Liu et al., 2022a; Kwon et al., 2023b). There is also another line of work that considered the setting of a non-strongly convex lower-level problem (Sabach & Shtern, 2017; Sow et al., 2022; Liu et al., 2020; Shen & Chen, 2023; Kwon et al., 2023a; Chen et al., 2023a). The most relevant work in this paper is (Hao et al., 2024), which considered the setting of unbounded smooth functions of the upper level and strongly convex functions of the lower level. However, their algorithm is not single-loop and requires extensive tuning efforts.

Unbounded Smoothness. The definition of relaxed smoothness was first proposed by (Zhang et al., 2020c), which was motivated by the loss landscape of recurrent neural networks. Zhang et al. (2020c) showed gradient clipping/normalization improves over gradient descent in this setting. Later, there is a line of work focusing on improved analysis (Zhang et al., 2020a; Jin et al., 2021), scalability (Liu et al., 2022b; Crawshaw et al., 2023a;b) and adaptive variants (Crawshaw et al., 2022; Faw et al., 2023; Wang et al., 2023; Li et al., 2023). Recently, some works studied the momentum and variance reduction techniques under individual relaxed smooth (Liu et al., 2023) or on-average relaxed smooth conditions (Reisizadeh et al., 2023) to achieve an improved convergence rate. People also studied various notions of relaxed smoothness, including coordinate-wise relaxed smoothness (Crawshaw et al., 2022), α-symmetric generalized smoothness (Chen et al., 2023c), and relaxed smoothness under the bilevel setting (Hao et al., 2024). This work considers the same problem setting as in (Hao et al., 2024), and focuses on the design of the single-loop algorithm.

Stochastic Optimization under Distributional Drift. Stochastic optimization with time drift is a classical topic and was studied extensively in the literature (Dupač, 1965; Guo & Ljung, 1995), mostly in the context of least squares problems and Kalman filtering. Recent work revisits these algorithms from the perspective of sequential stochastic/online optimization (Besbes et al., 2015; Wilson et al., 2018; Madden et al., 2021; Cutler et al., 2023). However, none of these works explores the connection between techniques in sequential optimization and the bilevel optimization problem as considered in this paper.

3. Preliminaries, Notations and Problem Setup

Define $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ as the inner product and Euclidean norm. Throughout the paper, we use asymptotic notation $\widetilde{O}(\cdot), \widetilde{\Theta}(\cdot), \widetilde{\Omega}(\cdot)$ to hide polylogarithmic factors in $1/\epsilon$ and $1/\delta$. Denote f: $\mathbb{R}^{d_x}\times\mathbb{R}^{d_y}\to\mathbb{R}$ as the upper-level function, and g: $\mathbb{R}^{d_x}\times\mathbb{R}^{d_y}\to\mathbb{R}$ as the lower-level function. The hypergradient $\nabla\Phi(x)$ shown in (Ghadimi & Wang, 2018) takes the form of

$$\nabla\Phi(x)=\nabla_x f(x,y^*(x))-\nabla^2_{xy}g(x,y^*(x))z^*(x), \quad (2)$$

where $z^*(x)$ is the solution to the linear system:

$$z^*(x)=\arg\min_z \frac{1}{2}\langle\nabla^2_{yy}g(x,y^*(x))z,z\rangle-\langle\nabla_y f(x,y^*(x)),z\rangle.$$

We make the following assumptions for the paper.

Assumption 3.1 ($(L_{x,0},L_{x,1},L_{y,0},L_{y,1})$-smoothness (Hao et al., 2024))**.** Let $w=(x,y)$ and $w'=(x',y')$, there exists $L_{x,0},L_{x,1},L_{y,0},L_{y,1}>0$ such that for all w,w', if $\|w-w'\|\le 1/\sqrt{2(L_{x,1}^2+L_{y,1}^2)}$, then

$$\|\nabla_x f(w)-\nabla_x f(w')\|\le(L_{x,0}+L_{x,1}\|\nabla_x f(w)\|)\|w-w'\|,$$
$$\|\nabla_y f(w)-\nabla_y f(w')\|\le(L_{y,0}+L_{y,1}\|\nabla_y f(w)\|)\|w-w'\|.$$

Remark: Assumption 3.1 is introduced by (Hao et al., 2024), and they empirically show that it is satisfied on recurrent neural networks with y being the last linear layer and x being previous layers. This assumption can be regarded as a generalization of the relaxed smoothness condition from single-level problems (Zhang et al., 2020c; Crawshaw et al., 2022) to the bilevel problem.

Assumption 3.2. Suppose the following holds for objective functions f and g: (i) For every x, $\|\nabla_y f(x,y^*(x))\|\le l_{f,0}$; (ii) For every x, $g(x,y)$ is μ-strongly convex in y for $\mu>0$; (iii) g is $l_{g,1}$-smooth jointly in (x,y); (iv) g is twice continuously differentiable, and $\nabla^2_{xy}g,\nabla^2_{yy}g$ are $l_{g,2}$-Lipschitz jointly in (x,y).

Remark: Assumption 3.2 is standard in the bilevel optimization literature (Kwon et al., 2023b; Hao et al., 2024; Ghadimi & Wang, 2018). In particular, Assumption 3.2 (i) is theoretically and empirically justified by (Hao et al., 2024) for recurrent neural networks. Under Assumptions 3.1 and 3.2, we can show that function $\Phi(x)$ satisfies standard relaxed smoothness condition: $\|\nabla\Phi(x)-\nabla\Phi(x')\|\le(L_0+L_1\|\nabla\Phi(x')\|)\|x-x'\|$ if x and x' are not far away from each other (Lemma C.3 in Appendix).

Assumption 3.3. The estimators used to calculate stochastic gradients and Hessian-vector products (i.e., $\nabla_x F(x,y;\xi)$,$\nabla_y F(x,y;\xi)$, $\nabla_y G(x,y;\xi)$, $\nabla^2_{xy}G(x,y;\zeta)$, $\nabla^2_{yy}G(x,y;\zeta)$) are *unbiased* and satisfy (assume $\lambda>0$):

$$\mathbb{E}_{\xi\sim\mathcal{D}_f}[\|\nabla_x F(x,y;\xi)-\nabla_x f(x,y)\|^2]\le\sigma^2_{f,1},$$
$$\mathbb{E}_{\xi\sim\mathcal{D}_f}[\|\nabla_y F(x,y;\xi)-\nabla_y f(x,y)\|^2]\le\sigma^2_{f,1},$$
$$\Pr\{\|\nabla_y G(x,y;\xi)-\nabla_y g(x,y)\|\ge\lambda\}\le 2\exp\left(-2\lambda^2/\sigma^2_{g,1}\right),$$
$$\mathbb{E}_{\zeta\sim\mathcal{D}_g}[\|\nabla^2_{xy}G(x,y;\zeta)-\nabla^2_{xy}g(x,y)\|^2]\le\sigma^2_{g,2},$$
$$\mathbb{E}_{\zeta\sim\mathcal{D}_g}[\|\nabla^2_{yy}G(x,y;\zeta)-\nabla^2_{yy}g(x,y)\|^2]\le\sigma^2_{g,2}.$$

Remark: Assumption 3.3 assumes that we have access to an unbiased stochastic gradient and a Hessian-vector product with bounded variance, which is standard in the literature (Ghadimi & Lan, 2013b; Ghadimi & Wang, 2018). We need the stochastic gradient of the lower-level problem to be light-tailed, which is also assumed by (Hao et al., 2024). This is a technical assumption that allows high probability analysis for y, which is a standard assumption in the optimization literature (Lan, 2012; Hazan & Kale, 2014).

4. Algorithm and Theoretical Analysis

4.1. Main Challenges and Algorithm Design

Main Challenges. We first illustrate why the existing work on bilevel optimization is insufficient for solving our problem with a single-loop algorithm. First, the analyses of single-loop bilevel algorithms for smooth bilevel optimization in the literature (i.e., the upper-level function has a Lipschitz gradient) (Hong et al., 2023; Dagréou et al., 2022; Chen et al., 2023a) typically design a single potential function to track progress in terms of both upper-level and lower-level variables and show that the potential function can decrease in expectation during algorithm updates. Their analysis is crucially based on L-smoothness of the upper-level function to control the approximation error of hypergradient. However, when the upper-level function is $(L_{x,0},L_{x,1},L_{y,0},L_{y,1})$-smooth, the bias in the hypergradient error depends on both the approximation error of the lower-level variable and the hypergradient in terms of the upper-level variable, which are statistically dependent and therefore cannot be analyzed easily using the standard expectation-based analysis for all variables. Second, the recent work of (Hao et al., 2024) addressed this issue by designing a new algorithm, along with a high probability analysis for the lower-level variable and a expectation-based analysis for the upper-level variable, but their algorithm

crucially relies on the nested-loop design. The algorithm in (Hao et al., 2024) has two components to update its lower-level variable: initialization refinement and periodic updates. Their key idea is to obtain an accurate estimator for the optimal lower-level variable at every iteration with high probability such that the hypergradient error can be controlled. However, such a strong requirement for the lower-level variable typically cannot be satisfied by any single-loop algorithms: single-loop algorithms typically make small progress on each step and can get an accurate solution after polynomial number of iterations, but cannot guarantee good estimates at every iteration.

Algorithm Design. To address these challenges, our algorithm design principle relies on the following important observation: obtaining accurate estimates for the lower-level variable at every iteration with high probability is only a sufficient condition for the algorithm in (Hao et al., 2024) to work, but it is not necessary: it is possible to establish a refined characterization and control of the lower-level variable which has weaker requirements than (Hao et al., 2024). This important insight makes the design of a single loop algorithm possible. The detailed description of our algorithm is illustrated in Algorithm 2. In particular, our algorithm first updates the lower-level variable by a few steps of SGD (line 3), and then simultaneously updates the upper-level variable by normalized SGD with momentum and the lower-level variable by SGD (line 4∼10). Our key novelty in the algorithm design and analysis comes from the novel perspective of connecting bilevel optimization and stochastic optimization under distribution drift: the procedure of updating the lower-level variable can be regarded as a stochastic optimization under distribution drift and the drift between iterations is small. In particular, the update rule in Algorithm 2 for warm-start (line 3) and y (line 7) can be viewed as a special case of SGD with distribution drift as specified in Algorithm 1, where the drift comes from the change of x and the change of x is small due to the normalization operator. Note that in the warm-start step (line 3) in Algorithm 2, x_0 is fixed and there is no distribution drift.

Difference between SLIP and (Hao et al., 2024). First, the work of (Hao et al., 2024) has the initialization refinement subroutine to obtain an accurate initial estimate for the optimal lower-level solution to the initial upper-level variable (that is, $y^*(x_0)$), which requires epoch-SGD (Hazan et al., 2015; Ghadimi & Lan, 2013a) for polynomial number of iterations. In contrast, our algorithm has a short warm-start stage (line 3), which runs SGD for only logarithmic number of iterations. Second, the work of (Hao et al., 2024) needs to periodically update its lower-level variable for polynomial number of iterations, but our algorithm only needs to simply run SGD for the lower-level variable at every iteration and our lower-level update is performed simultaneously with the upper-level update (line 4∼10).

Algorithm 1 SGD WITH DISTRIBUTIONAL DRIFT

1: **Input:** $\{\tilde{x}_t\}, \tilde{y}_0, \alpha, N$ # $\texttt{SGD-DD}(\{\tilde{x}_t\}, \tilde{y}_0, \alpha, N)$
2: **for** $t = 0, 1, \ldots, N-1$ **do**
3: Sample $\tilde{\pi}_t$ from distribution $\mathcal{D}_g$
4: $\tilde{y}_{t+1} = \tilde{y}_t - \alpha \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t)$
5: **end for**

Algorithm 2 SINGLE LOOP BILEVEL OPTIMIZER (SLIP)

1: **Input:** $\alpha^{\text{init}}, \alpha, \beta, \gamma, \eta, T_0, T$
2: **Initialize:** $x_0, y_0^{\text{init}}, z_0, m_0 = 0$
3: $y_0 = \texttt{SGD-DD}(\{x_0\}, y_0^{\text{init}}, \alpha^{\text{init}}, T_0)$ # Warm-start
4: **for** $t = 0, 1, \ldots, T-1$ **do**
5: Sample ξ_t, ξ_t' independently from distribution $\mathcal{D}_f$
6: Sample π_t, ζ_t, ζ_t' independently from distribution $\mathcal{D}_g$
7: $y_{t+1} = y_t - \alpha \nabla_y G(x_t, y_t; \pi_t)$
8: $z_{t+1} = z_t - \gamma[\nabla_{yy}^2 G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t)]$
9: $m_{t+1} = \beta m_t + (1-\beta)(\nabla_x F(x_t, y_t; \xi_t') - \nabla_{xy}^2 G(x_t, y_t; \zeta_t') z_t)$
10: $x_{t+1} = x_t - \eta \frac{m_{t+1}}{\|m_{t+1}\|}$
11: **end for**

4.2. Main Results

We first introduce a few notations. Define $\sigma(\cdot)$ as the σ-algebra generated by the random variables in the argument. We define the following filtrations: $\widetilde{\mathcal{F}}_t^1 = \sigma(\tilde{\pi}_0, \ldots, \tilde{\pi}_{t-1})$, $\mathcal{F}_t^1 = \sigma(\pi_0, \ldots, \pi_{t-1})$, $\mathcal{F}_t^2 = \sigma(\xi_0, \ldots, \xi_{t-1}, \zeta_0, \ldots, \zeta_{t-1})$, $\mathcal{F}_t^3 = \sigma(\xi_0', \ldots, \xi_{t-1}', \zeta_0', \ldots, \zeta_{t-1}')$, where $1 \le t \le T$. Define $\widetilde{\mathcal{F}}_t = \sigma(\mathcal{F}_t^2 \cup \mathcal{F}_t^3)$, $\mathcal{F}_t = \sigma(\widetilde{\mathcal{F}}_{T_0}^1 \cup \mathcal{F}_t^1 \cup \mathcal{F}_t^2 \cup \mathcal{F}_t^3)$. We use $\mathbb{E}_t$ and $\mathbb{E}$ to denote the conditional expectation $\mathbb{E}[\cdot \mid \widetilde{\mathcal{F}}_t]$ and the total expectation over $\widetilde{\mathcal{F}}_T$ respectively. In Algorithm 2, define $\widehat{\nabla}\Phi(x, y, z; \xi, \zeta) := \nabla_x F(x, y; \xi) - \nabla_{xy}^2 G(x, y; \zeta) z$ and thus we could write line 9 as $m_{t+1} = \beta m_t + (1-\beta)\widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t')$.

4.2.1. CONVERGENCE IN EXPECTATION

Theorem 4.1. *Suppose Assumptions 3.1 and 3.2 hold. Let $\{x_t\}$ be the iterates produced by Algorithm 2. For any given $\delta \in (0, 1)$ and sufficiently small ϵ (see the exact choice of ϵ in (D.37)), if we choose $\alpha^{\text{init}}, \alpha, \beta, \gamma, \eta, T_0$ as*

$$\alpha^{\text{init}} = \min\left\{\frac{1}{2l_{g,1}}, \frac{\mu}{2048 L_1^2 \sigma_{g,1}^2 \log(e/\delta)}\right\},$$

$$1-\beta = \Theta\left(\frac{\mu^2\epsilon^2}{L_0^2\sigma_{g,1}^2 \log^2(B)}\right), \eta = \frac{\mu\epsilon}{8 l_{g,1} L_0 \log(A)}(1-\beta),$$

$$\gamma = \frac{1-\beta}{\mu}, \quad \alpha = \frac{8(1-\beta)}{\mu}, \quad T_0 = \frac{\log\left(256 L_1^2 \|y_0^{\text{init}} - y_0^*\|^2\right)}{\log\left(2/(2-\mu\alpha^{\text{init}})\right)},$$

where Δ_0, A and B are defined in (D.13), then with probability at least $1-2\delta$ over the randomness in $\sigma(\widetilde{\mathcal{F}}_{T_0}^1 \cup \mathcal{F}_T^1)$, Algorithm 2 guarantees $\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla\Phi(x_t)\| \le 14\epsilon$ with

at most $T = \frac{4\Delta_0}{\eta\epsilon}$ *iterations, where the expectation is taken over over the randomness in* $\widetilde{\mathcal{F}}_T$.

Remark: The full specification and the proofs are included in Appendix (i.e., Theorem D.13). Theorem 4.1 shows that if $\eta = \widetilde{\Theta}(\epsilon^3)$, $1-\beta = \widetilde{\Theta}(\epsilon^2)$, $\alpha = \widetilde{\Theta}(\epsilon^2)$ and $\gamma = \widetilde{\Theta}(\epsilon^2)$, then Algorithm 2 converges to ϵ-stationary points within $\widetilde{O}(\epsilon^{-4})$ iterations in expectation. This complexity result matches the double-loop algorithm in previous work (Hao et al., 2024). In addition, in terms of the dependency on ϵ, our complexity bound is nearly optimal up to logarithmic factors due to the $\Omega(\epsilon^{-4})$ lower bound of nonconvex stochastic optimization for smooth single-level problems (Arjevani et al., 2023) when there is no mean-squared smooth condition of the stochastic gradient.

4.2.2. High Probability Guarantees

In this section, we provide a high probability result of our algorithm, which requires the following assumption.

Assumption 4.2. The estimators used to calculate stochastic gradients and Hessian-vector products (i.e., $\nabla_x F(x,y;\xi)$,$\nabla_y F(x,y;\xi)$, $\nabla_y G(x,y;\xi)$, $\nabla^2_{xy} G(x,y;\zeta)$, $\nabla^2_{yy} G(x,y;\zeta)$) are *unbiased* and satisfy (assume $\lambda > 0$):

$$\begin{aligned}
&\forall \xi,\ \|\nabla_x F(x,y;\xi) - \nabla_x f(x,y)\| \le \sigma_{f,1},\\
&\forall \xi,\ \|\nabla_y F(x,y;\xi) - \nabla_y f(x,y)\| \le \sigma_{f,1},\\
&\Pr\{\|\nabla_y G(x,y;\xi) - \nabla_y g(x,y)\| \ge \lambda\} \le 2\exp\left(-2\lambda^2/\sigma^2_{g,1}\right),\\
&\forall \zeta,\ \|\nabla^2_{xy} G(x,y;\zeta) - \nabla^2_{xy} g(x,y)\| \le \sigma_{g,2},\\
&\forall \zeta, z,\ \|(\nabla^2_{yy} G(x,y;\zeta) - \nabla^2_{yy} g(x,y))z\| \le \sigma_z.
\end{aligned}$$

Remark: Assumption 4.2 is a technical assumption to establish the high probability result. It assumes that the estimators either have almost sure bounded noise or light-tailed noise. Similar assumptions have been made in the literature on optimization for relaxed smooth functions (Zhang et al., 2020b;a; Liu et al., 2023) and strongly convex functions (Cutler et al., 2023). A more in-depth discussion is included in Appendix E.1. Our setting is more challenging because their goal is to optimize a single-level relaxed smooth function, while our work is for bilevel optimization under unbounded smoothness.

Theorem 4.3. *Suppose Assumptions 3.1 and 4.2 hold. Let* $\{x_t\}$ *be the iterates produced by Algorithm 2. For any given* $\delta \in (0,1)$ *and sufficiently small* ϵ *(see exact choice of* ϵ *in (E.9)), if we choose* $\gamma = \frac{16}{\mu}(1-\beta)$*, and the same* $\alpha^{\text{init}}, \alpha, \beta, \eta, T_0$ *as in Theorem 4.1, then with probability at least* $1-4\delta$ *over the randomness in* $\mathcal{F}_T$*, Algorithm 2 guarantees* $\frac{1}{T}\sum_{t=0}^{T}\|\nabla\Phi(x_t)\| \le \epsilon$ *with at most* $T = \frac{4\Delta_0}{\eta\epsilon}$ *iterations.*

Remark: Theorem 4.3 establishes a high probability result of bilevel optimization under unbounded smoothness. Note that the choice of η is the same as in Theorem 4.1 (i.e., $\eta = \widetilde{\Theta}(\epsilon^3)$), therefore we can get $\widetilde{O}(\epsilon^{-4})$ iteration complexity to find an ϵ-stationary point with high probability. To the best our knowledge, this is the first high probability convergence guarantee for stochastic bilevel optimization.

4.3. Proof Sketch

In this section, we mainly provide a proof sketch of Theorem 4.1 and briefly discuss the high probability proof of Theorem 4.3. The detailed proofs can be found in Appendix D and E, respectively. Define $y_t^* = y^*(x_t), z_t^* = z^*(x_t), \tilde{y}_t^* = y^*(\tilde{x}_t)$. Similar to the previous work (Hao et al., 2024), it is difficult for us to handle the hypergradient bias term $\mathbb{E}_t[\|\widehat{\nabla}\Phi(x_t,y_t,z_t;\xi_t',\zeta_t') - \nabla\Phi(x_t)\|]$: the upper bound of this quantity depends on $L_{x,1}\|y_t - y_t^*\|\|\nabla\Phi(x_t)\|$ due to Assumption 3.1. The work of (Hao et al., 2024) uses a double-loop procedure to ensure that $\|y_t - y_t^*\|$ is very small (that is, the same order as ϵ) for every t with high probability, which is too demanding and cannot hold for our proposed single-loop algorithm.

To address this issue, the key idea is that we do not require $\|y_t - y_t^*\|$ to be small for every t, instead we only need $\|y_t - y_t^*\|$ to be smaller than some constant (i.e, $\frac{1}{8L_1}$) for every t and the weighted average of $\|y_t - y_t^*\|$ over all iterations is smaller than ϵ. In this way, we can also handle the hypergradient bias and establish the convergence. To this end, we introduce the following lemmas. Lemma 4.4 is introduced to handle the approximation error of the lower-level variable for any slowly-changing upper-level sequences. Then we apply this lemma to the warm-start stage (line 3 in Algorithm 2) and the stage of simultaneous updates for lower-level and upper-level variables (line 4∼10), which ends up with Lemma 4.5 and Lemma 4.6 respectively.

Lemma 4.4. *Suppose Assumption 3.2 holds, let* $\{\tilde{y}_t\}$ *be the iterates produced by Algorithm 1 with any fixed input sequence* $\{\tilde{x}_t\}$ *such that* $\|\tilde{x}_{t+1} - \tilde{x}_t\| \le R$ *for all* $t \ge 0$*, and constant learning rate* $\alpha \le 1/(2l_{g,1})$*. Then for any fixed* $t \in [N]$ *and* $\delta \in (0,1)$*, the following estimate holds with probability at least* $1-\delta$ *over the randomness in* $\widetilde{\mathcal{F}}_t^1$*:*

$$\|\tilde{y}_t - \tilde{y}_t^*\|^2 \le \left(1 - \frac{\mu\alpha}{2}\right)^t \|\tilde{y}_0 - \tilde{y}_0^*\|^2 + \left[\frac{8\alpha\sigma^2_{g,1}}{\mu} + \frac{4R^2 l^2_{g,1}}{\mu^4\alpha^2}\right]\log\frac{e}{\delta},$$

where e *denotes the base of natural logarithms. As a consequence, for all* $t \in [N]$ *and* $\delta \in (0,1)$*, the following holds with probability at least* $1-\delta$ *over the randomness in* $\widetilde{\mathcal{F}}_N^1$*:*

$$\|\tilde{y}_t - \tilde{y}_t^*\|^2 \le \left(1 - \frac{\mu\alpha}{2}\right)^t \|\tilde{y}_0 - \tilde{y}_0^*\|^2 + \left[\frac{8\alpha\sigma^2_{g,1}}{\mu} + \frac{4R^2 l^2_{g,1}}{\mu^4\alpha^2}\right]\log\frac{eN}{\delta}.$$

Remark: Lemma 4.4 establishes the error of the lower-level problem at every iteration for *any fixed slowly changing upper-level sequence* $\{\tilde{x}_t\}$ with high probability. This lemma is a generalization of the techniques of stochastic optimization under distribution drift (e.g., Theorem 6 in (Cutler

et al., 2023)). It will be applied to two stages of our algorithm (warm-start stage in line 3, and simultaneous update stage in line 4∼10) to control the lower-level error.

Lemma 4.5 (Warm-start). *Suppose Assumption 3.2 holds and given any $\delta \in (0,1)$, let $\{y_t^{\mathsf{init}}\}$ be the iterates produced by Algorithm 1 starting from y_0^{init} with $R = 0$ (where R is defined in Lemma 4.4, it means that $\tilde{x}_t = x_0$ for any t). Under the same choice of learning rate α^{init} as in Theorem 4.1 and run Algorithm 1 for $T_0 = \frac{\log\left(256L_1^2\|y_0^{\mathsf{init}} - y_0^*\|^2\right)}{\log(2/(2-\mu\alpha^{\mathsf{init}}))} = \widetilde{O}(1)$ iterations, Algorithm 1 guarantees $\|y_{T_0}^{\mathsf{init}} - y_0^*\| \le 1/(8\sqrt{2}L_1)$ with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_{T_0}^1$ (we denote this event as $\mathcal{E}_{\mathsf{init}}$).*

Remark: Lemma 4.5 shows that it requires at most a logarithmic number of iterations in the warm-start stage to get constant error of the lower-level variable, with high probability. This lemma is an application of Lemma 4.4 with $R = 0$ since the upper-level variable is fixed to be x_0.

Lemma 4.6. *Under assumptions 3.1, 3.2 and event $\mathcal{E}_{\mathsf{init}}$, for any given $\delta \in (0,1)$ and sufficiently small ϵ (see exact choice of ϵ in (D.37)), under the same parameter choice as in Theorem 4.1 and run Algorithm 2 for $T = \frac{4\Delta_0}{\eta\epsilon}$ iterations. Then for all $t \in [T]$, Algorithm 2 guarantees with probability at least $1-\delta$ over the randomness in $\mathcal{F}_T^1$ (we denote this event as $\mathcal{E}_y$) that:*

(i) $\|y_t - y_t^*\| \le \frac{1}{8L_1}$,

(ii) $\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|y_i - y_i^*\| \le \frac{3}{32L_0}\epsilon$,

where Δ_0 is defined in (D.13).

Remark: Lemma 4.6 provides a refined characterization of the lower-level error during the simultaneous update stage (line 4∼10 in Algorithm 2): the error of every iterate of y is bounded by a constant and the weighted error of iterates of y is small. Another important aspect of this lemma is that the statement holds with high probability over $\mathcal{F}_T^1$, which is *independent of the randomness in x and z* since $\mathcal{F}_T^1$ is independent of $\mathcal{F}_t^2$ and $\mathcal{F}_t^3$. This nice property is crucial for our subsequent analysis of the sequence $\{x_t\}$ and $\{z_t\}$ without worrying about the dependency issue on filtrations.

Lemma 4.7. *Under Assumptions 3.1, 3.2 and events $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, define $\epsilon_t = m_{t+1} - \nabla\Phi(x_t)$ as the moving-average hypergradient estimation error. Then we have*

$$\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right] \le \left(\frac{\eta L_1\beta}{1-\beta} + \frac{1}{8}\right)\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| + O\left(\frac{T\eta L_0\beta}{1-\beta}\right.$$
$$+T\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2 + \frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2} + l_{g,1}\sqrt{T}\sqrt{\sum_{t=0}^{T-1}\mathbb{E}[\|z_t - z_t^*\|^2]}$$
$$\left.+\frac{\|\nabla\Phi(x_0)\| + L_0\Delta_{y,0}}{1-\beta} + L_0 T\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\frac{eT}{\delta}}\right)$$

where $\Delta_{y,0}$ is defined in (D.13), and the expectation is taken over the randomness in $\widetilde{\mathcal{F}}_T$.

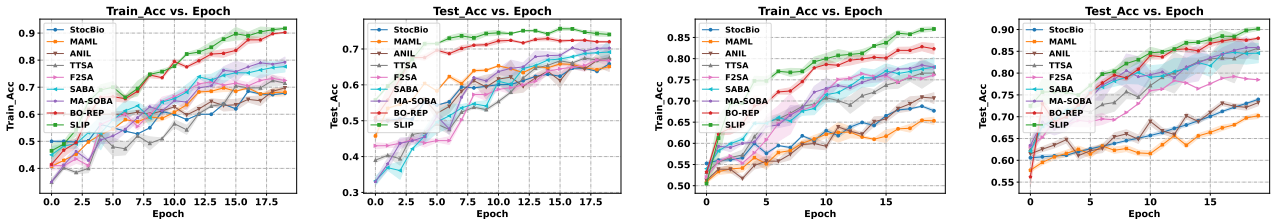
Remark: This lemma provides an upper bound for the cumulative hypergradient estimation error over T iterations. Under the parameter setting of Theorem 4.1, we can see that the RHS can be divided into two parts. The first part is the summation of history gradient norm, which can be dominated by a negative gradient term in the descent lemma. The second part consists of several error terms that grow sublinearly in terms of T (because the averaged expected squared error for variable z can be shown to grow sublinearly in T as well). The fact that these error terms vanish during the optimization process is crucial to establish the convergence result.

Proof Sketch of High Probability Guarantees in Theorem 4.3. The full proof is included in Appendix E and we provide the roadmap of the proof here. The main difference between high probability guarantees and expectation guarantees is that we also need to provide a high probability analysis for variables z and x. The idea of the proof is to build on the high probability result in y and gradually establish high probability results for all variables. In particular, from Lemma 4.5, we know that the event $\mathcal{E}_{\mathsf{init}}$ happens with probability $1-\delta$. Then from Lemma 4.6, we know that with probability at least $1-\delta$, we have $\mathcal{E}_y$ happening, provided that $\mathcal{E}_{\mathsf{init}}$ happens. The next step is to show under events $\mathcal{E}_y$ and $\mathcal{E}_{\mathsf{init}}$, we have a nice bound for z with probability $1-\delta$ (this is proved in Lemma E.4 in Appendix E, and this event is denoted as $\mathcal{E}_z$). Then under $\mathcal{E}_{\mathsf{init}}$, $\mathcal{E}_y$ and $\mathcal{E}_z$, we can obtain good hypergradient error with probability $1-\delta$ (this is proved in Lemma E.6 in Appendix E, and this event is denoted as $\mathcal{E}_x$) and derive the final convergence result. Therefore, by the rule of conditional probability, we show that the good event (i.e., $\mathcal{E}_x \cap \mathcal{E}_y \cap \mathcal{E}_z \cap \mathcal{E}_{\mathsf{init}}$) happens with probability $(1-\delta)^4 \ge 1 - 4\delta$ for $\delta \in (0,1)$.

5. Experiments

5.1. Hyper-representation Learning

In this section, we conduct experiments on an important application of bilevel optimization with unbounded smooth upper-level function: hyper-representation learning for text classification (i.e., meta-learning). The goal of hyper-representation learning is to try to find a good model representation parameterized by x, such that it can quickly adapt to new tasks by quickly tuning the task-specific parameter y_i by a few steps of gradient updates. The learning procedure can be formally characterized by bilevel optimization (Ji et al., 2021; Hao et al., 2024). We define a sequence of K tasks, which consists of the training set $\{\mathcal{D}_i^{tr} \mid i = 1, ..., K\}$ and validation set $\{\mathcal{D}_i^{val} \mid i = 1, ..., K\}$. The loss function of the model on a uniformly sampled data set $\xi_i \sim D_i^{tr}$ can



(a) Training accuracy on SNLI (b) Test accuracy on SNLI (c) Training accuracy on ARD (d) Test accuracy on ARD

Figure 1. Comparison with bilevel optimization baselines on Hyper-representation. Figure (a) and (b) are the results in the SNLI dataset. Figures (c) and (d) are the results of the Amazon Review Dataset (ARD).

be denoted as $\mathcal{L}(x, y_i; \xi_i)$.

From the perspective of bilevel optimization, the lower level function aims to find an optimal task-specific parameter y_i^* on the training set $\mathcal{D}_i^{tr}$, given the meta parameter x. The upper-level function evaluates each $y_i^*, i = 1, ..., K$ on the corresponding validation set $\mathcal{D}_i^{val}$ and leverages all the gradient information to update the meta-parameter x. If we denote $y = (y_1, y_2, ..., y_K)$, we can formalize this problem as follows:

$$\min_x \frac{1}{K}\sum_{i=1}^{K}\frac{1}{|\mathcal{D}_i^{val}|}\sum_{\xi\in\mathcal{D}_i^{val}}\mathcal{L}(x, y^*(x); \xi), \text{ s.t.,}$$
$$y^*(x) = \arg\min_y \frac{1}{K}\sum_{i=1}^{K}\frac{1}{|\mathcal{D}_i^{tr}|}\sum_{\zeta\in\mathcal{D}_i^{tr}}\mathcal{L}(x, y_i; \zeta) + \frac{\mu}{2}\|y_i\|^2, \tag{3}$$

where the lower-level function contains a regularizer $\frac{\mu}{2}\|y_i\|^2$, which ensures that the lower-level function is strongly convex. In practice, a fully-connected layer is widely used as a classifier y for the meta-learning model (Bertinetto et al., 2018; Ji et al., 2021), and a different classifier y_i will be used for a specific task. The remaining layers consist of recurrent neural layers, which are used to learn from natural language data. Therefore, the lower-level function satisfies the μ-strongly convex assumption, and the upper-level function satisfies the nonconvex and unbounded smoothness assumptions (i.e., Assumption 3.1).

We performed the hyper-representation learning experiment for text classification task on Stanford Natural Language Inference (SNLI) (Bowman et al., 2015) and Amazon Review Dataset (ARD) (Blitzer et al., 2006) datasets. SNLI is a naturalistic corpus of 570k pairs of sentences labeled "entailment", "contradiction", and "independence". We construct $K = 25$ tasks, where $\mathcal{D}_i^{tr}$ and $\mathcal{D}_i^{val}$ randomly sample two disjoint categories from the original data, respectively. ARD provides positive and negative customer reviews for 25 types of products. The following experimental setup is from (Hao et al., 2024): we choose three types (i.e., office products, automotive, and computer video games) as the validation set and the remaining types as the training set. The number of samples in training set and validation set keeps the same. In training phase, a good hyper-representation x is obtained, and will be used in the test phase for performance evaluation. Note that x is fixed during the test phase, and only task-specific parameter y is fine-tuned for a quick adaptation.

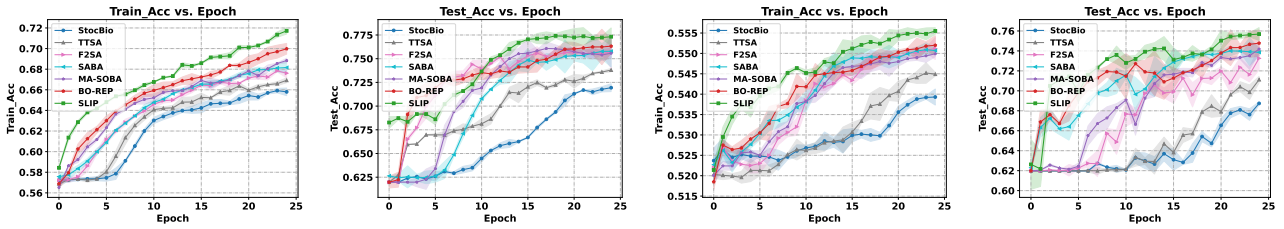
We compare our algorithm SLIP with other baselines, including typical meta-learning algorithms MAML (Rajeswaran et al., 2019) and ANIL (Raghu et al., 2019), bilevel optimization method: StocBio (Ji et al., 2021), TTSA (Hong et al., 2023), F^{2}SA (Kwon et al., 2023b), SABA (Dagréou et al., 2022), MA-SOBA (Chen et al., 2023a), and BO-REP (Hao et al., 2024). For a fair comparison, we use a 2-layer recurrent neural network as the representation layers with input dimension=300, hidden dimension=4096 and output dimension=512. A fully-connected layer is used as the classifier with its output dimention set to 2 on SNLI and 3 on ARD.

We run every method for 20 epochs, with minibatch size 50 for both training and validation set. Within each epoch, we run 500 iterations of our algorithm to traverse the constructed training and validation task on SNLI, and 400 iterations on ARD. The evolution of the training and test accuracy is shown in Figure 1 [2]. We can see that our proposed SLIP algorithm consistently outperforms all other baselines. More details can be found in Appendix F.1.

5.2. Data Hyper-cleaning

It is common that the data can be corrupted by noise, which presents a challenge to train a model at a certain level of

[2]Here an epoch means a full pass over the validation set, i.e., for upper-level variable x update. For a more comprehensive comparison, we re-conduct experiments where an epoch means a full pass over the training set, i.e., for lower-level variable y update. In this case there are fewer x updates than the previous run due to the warm-start phase. The results show that SLIP is still empirically better than other baselines. See Appendix G for more details.



(a) Training acc with $p = 0.2$ (b) Test acc with $p = 0.2$ (c) Training acc with $p = 0.4$ (d) Test acc with $p = 0.4$

Figure 2. Comparison with bilevel optimization baselinses on data hyper-cleaning. Figure (a), (b) are the results with the corruption rate $p = 0.2$. Figure (c), (d) are the results with the corruption rate $p = 0.4$.

noise. The data hyper-cleaning task (Shaban et al., 2019) aims to train a model on a corrupted set. An important approach to solving this problem is to learn a weight for each individual sample so that the weights associated with the noisy data will be assigned a small magnitude value by training. We typically solve the data hyper-cleaning task by bilevel optimization. Given an initial weight $x_i \in \mathbb{R}$ for each sample, a DNN model with parameter $y \in \mathbb{R}^d$, loss function $\mathcal{L}(\cdot)$, a corrupted training set $\mathcal{D}^{tr}$ and a non-corrupted validation set $\mathcal{D}^{val}$, the model is trained on the weighted set $\mathcal{D}^{tr}$ and tries to achieve good performance on the non-corrupted validation set $\mathcal{D}^{val}$. Formally, this bilevel optimization can be formulated as

$$\begin{aligned} &\min_x \frac{1}{|\mathcal{D}^{val}|} \sum_{\xi \in \mathcal{D}^{val}} \mathcal{L}(y^*(x); \xi), \\ &s.t.,\ y^*(x) = \arg\min_y \frac{1}{|\mathcal{D}^{tr}|} \sum_{\zeta_i \in \mathcal{D}^{tr}} \sigma(x_i)\mathcal{L}(y; \zeta_i) + \lambda \|y\|^2, \end{aligned} \tag{4}$$

where the activation function $\sigma(z) = \frac{1}{1+e^{-z}}$ and the parameter $\lambda > 0$ is the regularizer factor.

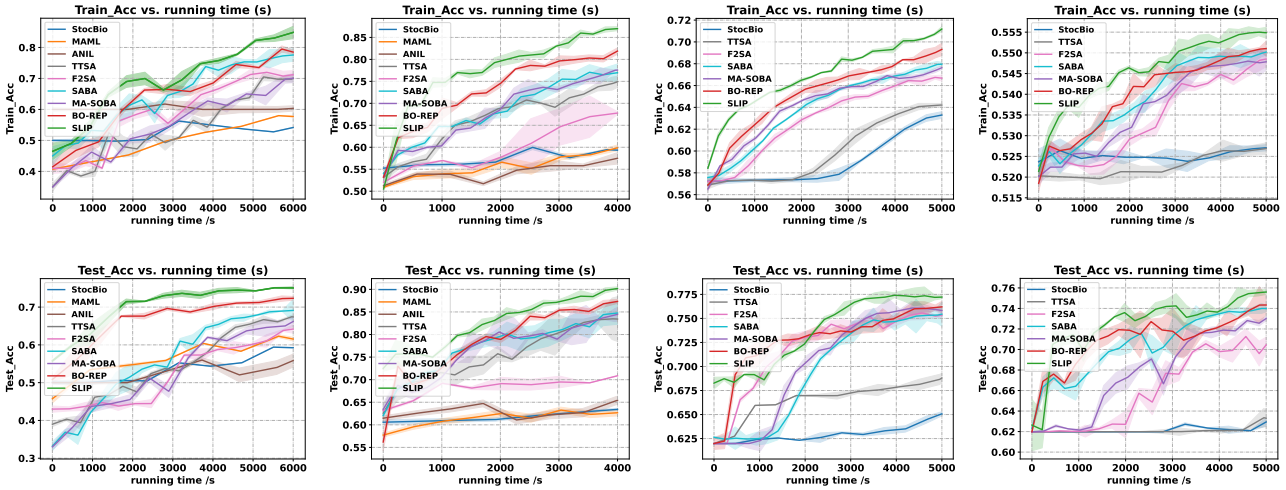
We performed the data hyper-cleaning experiment on Sentiment140 (Go et al., 2009) for text classification. The Sentiment140 dataset provides a large corpus of tweets that have been automatically labeled with sentiment (positive or negative) based on the presence of emoticons. This data set is commonly used as noisy labels for sentiment classification. We preprocess the original training set by randomly sampling a certain proportion p of data labels and flipping them, where p is called the corruption rate. In this paper, we set the corruption rate to 0.2 and 0.4.

For all baselines, we adopt a 2-layer recurrent neural network which is the same as that mentioned in Section 5.1 with the input dimension = 300 and the output dimension=2. The sample weight x_i is uniformly initialized to 1.0. We compare SLIP with other bilevel optimization algorithms, including StocBio (Ji et al., 2021), TTSA (Hong et al., 2023), F^{2}SA (Kwon et al., 2023b), SABA (Dagréou et al., 2022), MA-SOBA (Chen et al., 2023a), and BO-REP (Hao et al., 2024). We ran all the methods for 25 epochs, and it required 63 iterations within each epoch to traverse the training and validation set with minibatch size = 512.

The experimental results are presented in Figure 2, where (a) and (b) show the results with the corruption rate $p = 0.2$ and Figure (c) and (d) show the results with the corruption rate $p = 0.4$. Our SLIP outperforms all the baselines consistently, which demonstrates the effectiveness and superiority of our algorithm in data hyper-cleaning compared with others. More experimental details can be found in Appendix F.2.

5.3. Running Time Comparison

To evaluate the practical efficiency of various bilevel algorithms, we follow (Ji et al., 2021; Dagréou et al., 2022) to compare the performance of the baselines over the running time. Each algorithm runs separately on a single device with a NVIDIA RTX 6000 graphics card and an AMD EPYC 7513 32-Core Processor, while the training/test accuracy of the corresponding time is recorded. We show the results of accuracy versus running time in Figure 3. In particular, Figures (a), (b) show the experimental results of hyper-representation learning on SNLI and ARD datasets, respectively. Figures (c), (d) show the results of data hyper-cleaning in the Sentiment140 data set, where the corruption rate is $p = 0.2$ in (c) and $p = 0.4$ in (d). Note that the same time scale is used for one figure for a fair comparison. We can observe that SLIP runs faster than other baselines in all the figures, and BO-REP follows. One interesting observation is that in the hyper-representation learning experiment, SLIP runs significantly faster than BO-REP (Hao et al., 2024) (e.g., in Figure 3 (a), (b)). The reason is due to the single-loop nature of SLIP algorithm: when calculating the gradient of the previous layers' parameter (i.e., x in (3)), the gradient of the last layer's parameter (i.e., y in (3)) is automatically calculated in the same backpropagation so that we do not need to recalculate it again for updating y, so it saves running time compared to BO-REP. In contrast,



(a) Hyper-representation (SNLI) (b) Hyper-representation (ARD) (c) Data Hyper-cleaning (p=0.2) (d) Data Hyper-cleaning (p=0.4)

Figure 3. Comparison on running time. (a) Results of Hyper-representation on SNLI dataset. (b) Results of Hyper-representation on Amazon Review Dataset (ARD). (c), (d) Results of data Hyper-cleaning on Sentiment140 with corruption rate $p = 0.2$ and $p = 0.4$.

BO-REP needs multiple separate calculations of gradient w.r.t. y during its periodic updates for y.

6. Conclusion

In this paper, we studied the problem of stochastic bilevel optimization, where the upper-level function has potential unbounded smoothness and the lower-level problem is strongly convex. We have designed a single loop algorithm with $\widetilde{O}(1/\epsilon^4)$ oracle calls to find an ϵ-stationary point, where each oracle call invokes a stochastic gradient or a Hessian-vector product. Our complexity bounds hold both in expectation and with high probability, under different assumptions. Our bound is nearly optimal due to the lower bounds of non-convex stochastic optimization for single-level problems, when there is no mean squared smoothness of the stochastic gradient oracles (Arjevani et al., 2023). In the future, we plan to further improve the complexity bounds for bilevel optimization under stronger assumptions. For example, one interesting question to investigate is to obtain $O(1/\epsilon^3)$ complexity under the individual relaxed smoothness assumption for the stochastic gradient/Hessian-vector product oracle.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. This work has been supported by a grant from George Mason University, the Presidential Scholarship from George Mason University, a ORIEI seed funding from George Mason University, and a Cisco Faculty Research Award. The Computations were run on ARGO, a research computing cluster provided by the Office of Research Computing at George Mason University (URL: https://orc.gmu.edu).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Anandalingam, G. and White, D. A solution method for the linear static stackelberg problem using penalty functions. *IEEE Transactions on automatic control*, 35(10):1170–1173, 1990.

Arbel, M. and Mairal, J. Amortized implicit differentiation for stochastic bilevel optimization. *arXiv preprint arXiv:2111.14580*, 2021.

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199 (1-2):165–214, 2023.

Bertinetto, L., Henriques, J. F., Torr, P. H., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

Besbes, O., Gur, Y., and Zeevi, A. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Blitzer, J., McDonald, R., and Pereira, F. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pp. 120–128, 2006.

Borsos, Z., Mutny, M., and Krause, A. Coresets via bilevel optimization for continual learning and streaming. *Advances in neural information processing systems*, 33: 14879–14890, 2020.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*, 2015.

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

Chen, L., Xu, J., and Zhang, J. On bilevel optimization without lower-level strong convexity. *arXiv preprint arXiv:2301.00712*, 2023a.

Chen, T., Sun, Y., and Yin, W. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. *Advances in Neural Information Processing Systems*, 34:25294–25307, 2021.

Chen, T., Sun, Y., Xiao, Q., and Yin, W. A single-timescale method for stochastic bilevel optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 2466–2488. PMLR, 2022.

Chen, X., Xiao, T., and Balasubramanian, K. Optimal algorithms for stochastic bilevel optimization under relaxed smoothness conditions. *arXiv preprint arXiv:2306.12067*, 2023b.

Chen, Z., Zhou, Y., Liang, Y., and Lu, Z. Generalized-smooth nonconvex optimization is as efficient as smooth nonconvex optimization. *arXiv preprint arXiv:2303.02854*, 2023c.

Crawshaw, M., Liu, M., Orabona, F., Zhang, W., and Zhuang, Z. Robustness to unbounded smoothness of generalized signsgd. *Advances in neural information processing systems*, 2022.

Crawshaw, M., Bao, Y., and Liu, M. Episode: Episodic gradient clipping with periodic resampled corrections for federated learning with heterogeneous data. In *The Eleventh International Conference on Learning Representations*, 2023a.

Crawshaw, M., Bao, Y., and Liu, M. Federated learning with client subsampling, data heterogeneity, and unbounded smoothness: A new algorithm and lower bounds. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Cutler, J., Drusvyatskiy, D., and Harchaoui, Z. Stochastic optimization under distributional drift. *Journal of Machine Learning Research*, 24(147):1–56, 2023.

Dagréou, M., Ablin, P., Vaiter, S., and Moreau, T. A framework for bilevel optimization that enables stochastic and global variance reduction algorithms. *Advances in Neural Information Processing Systems*, 35:26698–26710, 2022.

Dagréou, M., Moreau, T., Vaiter, S., and Ablin, P. A lower bound and a near-optimal algorithm for bilevel empirical risk minimization. *arXiv preprint arXiv:2302.08766*, 2023.

Defazio, A., Bach, F., and Lacoste-Julien, S. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pp. 1646–1654, 2014.

Dupač, V. A dynamic stochastic approximation method. *The Annals of Mathematical Statistics*, pp. 1695–1702, 1965.

Elman, J. L. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

Faw, M., Rout, L., Caramanis, C., and Shakkottai, S. Beyond uniform smoothness: A stopped analysis of adaptive sgd. *arXiv preprint arXiv:2302.06570*, 2023.

Feurer, M. and Hutter, F. Hyperparameter optimization. *Automated machine learning: Methods, systems, challenges*, pp. 3–33, 2019.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., and Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. In *International conference on machine learning*, pp. 1568–1577. PMLR, 2018.

Ghadimi, S. and Lan, G. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013a.

Ghadimi, S. and Lan, G. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013b.

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

Go, A., Bhayani, R., and Huang, L. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009.

Grazzi, R., Pontil, M., and Salzo, S. Bilevel optimization with a lower-level contraction: Optimal sample complexity without warm-start. *arXiv preprint arXiv:2202.03397*, 2022.

Guo, L. and Ljung, L. Exponential stability of general tracking algorithms. *IEEE Transactions on Automatic Control*, 40(8):1376–1387, 1995.

Guo, Z., Hu, Q., Zhang, L., and Yang, T. Randomized stochastic variance-reduced methods for multi-task stochastic bilevel optimization. *arXiv preprint arXiv:2105.02266*, 2021.

Hao, J., Ji, K., and Liu, M. Bilevel coreset selection in continual learning: A new formulation and algorithm. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Hao, J., Gong, X., and Liu, M. Bilevel optimization under unbounded smoothness: A new algorithm and convergence analysis. In *The Twelfth International Conference on Learning Representations*, 2024.

Hazan, E. and Kale, S. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *Journal of Machine Learning Research*, 15 (1):2489–2512, 2014.

Hazan, E., Levy, K. Y., and Shalev-Shwartz, S. Beyond convexity: Stochastic quasi-convex optimization. *arXiv preprint arXiv:1507.02030*, 2015.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Hong, M., Wai, H.-T., Wang, Z., and Yang, Z. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1):147–180, 2023.

Ji, K., Yang, J., and Liang, Y. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*, pp. 4882–4892. PMLR, 2021.

Jin, J., Zhang, B., Wang, H., and Wang, L. Non-convex distributionally robust optimization: Non-asymptotic analysis. *Advances in Neural Information Processing Systems*, 34:2771–2782, 2021.

Khanduri, P., Zeng, S., Hong, M., Wai, H.-T., Wang, Z., and Yang, Z. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. *Advances in neural information processing systems*, 34:30271–30283, 2021.

Konda, V. and Tsitsiklis, J. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

Kwon, J., Kwon, D., Wright, S., and Nowak, R. On penalty methods for nonconvex bilevel optimization and first-order stochastic approximation. *arXiv preprint arXiv:2309.01753*, 2023a.

Kwon, J., Kwon, D., Wright, S., and Nowak, R. D. A fully first-order method for stochastic bilevel optimization. In *International Conference on Machine Learning*, pp. 18083–18113. PMLR, 2023b.

Lan, G. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1-2):365–397, 2012.

Li, H., Jadbabaie, A., and Rakhlin, A. Convergence of adam under relaxed assumptions. *arXiv preprint arXiv:2304.13972*, 2023.

Liu, B., Ye, M., Wright, S., Stone, P., and Liu, Q. Bome! bilevel optimization made easy: A simple first-order approach. *Advances in Neural Information Processing Systems*, 35:17248–17262, 2022a.

Liu, M., Zhuang, Z., Lei, Y., and Liao, C. A communication-efficient distributed gradient clipping algorithm for training deep neural networks. *Advances in Neural Information Processing Systems*, 35:26204–26217, 2022b.

Liu, R., Mu, P., Yuan, X., Zeng, S., and Zhang, J. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *International Conference on Machine Learning (ICML)*, 2020.

Liu, Z., Jagabathula, S., and Zhou, Z. Near-optimal nonconvex stochastic optimization under generalized smoothness. *arXiv preprint arXiv:2302.0603*, 2023.

Madden, L., Becker, S., and Dall'Anese, E. Bounds for the tracking error of first-order online optimization methods. *Journal of Optimization Theory and Applications*, 189: 437–457, 2021.

Raghu, A., Raghu, M., Bengio, S., and Vinyals, O. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*, 2019.

Rajeswaran, A., Finn, C., Kakade, S. M., and Levine, S. Meta-learning with implicit gradients. *Advances in neural information processing systems*, 32, 2019.

Reisizadeh, A., Li, H., Das, S., and Jadbabaie, A. Variance-reduced clipping for non-convex optimization. *arXiv preprint arXiv:2303.00883*, 2023.

Roh, Y., Lee, K., Whang, S. E., and Suh, C. Fairbatch: Batch selection for model fairness. *arXiv preprint arXiv:2012.01696*, 2020.

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

Shaban, A., Cheng, C.-A., Hatch, N., and Boots, B. Truncated back-propagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1723–1732. PMLR, 2019.

Shen, H. and Chen, T. On penalty-based bilevel gradient descent method. *arXiv preprint arXiv:2302.05185*, 2023.

Sow, D., Ji, K., Guan, Z., and Liang, Y. A constrained optimization approach to bilevel optimization with multiple inner minima. *arXiv preprint arXiv:2203.01123*, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vicente, L., Savard, G., and Júdice, J. Descent approaches for quadratic bilevel programming. *Journal of optimization theory and applications*, 81(2):379–399, 1994.

Wang, B., Zhang, H., Ma, Z., and Chen, W. Convergence of adagrad for non-convex objectives: Simple proofs and relaxed assumptions. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 161–190. PMLR, 2023.

White, D. J. and Anandalingam, G. A penalty function approach for solving bi-level linear programs. *Journal of Global Optimization*, 3:397–419, 1993.

Wilson, C., Veeravalli, V. V., and Nedić, A. Adaptive sequential stochastic optimization. *IEEE Transactions on Automatic Control*, 64(2):496–509, 2018.

Yang, J., Ji, K., and Liang, Y. Provably faster algorithms for bilevel optimization. *Advances in Neural Information Processing Systems*, 34:13670–13682, 2021.

Zhang, B., Jin, J., Fang, C., and Wang, L. Improved analysis of clipping algorithms for non-convex optimization. *Advances in Neural Information Processing Systems*, 2020a.

Zhang, J., He, T., Sra, S., and Jadbabaie, A. Why gradient clipping accelerates training: A theoretical justification for adaptivity. *International Conference on Learning Representations*, 2020b.

Zhang, J., He, T., Sra, S., and Jadbabaie, A. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations*, 2020c.

A. Comparison Table of Stochastic Bilevel Optimization Algorithms

Table 1. Comparison of stochastic bilevel optimization algorithms in the nonconvex-strongly-convex setting under different smoothness assumptions on f and g. The oracle complexity stands for the number of oracle calls to stochastic gradients and stochastic Hessian/Jacobian-vector products to find an ϵ-stationary point. $\mathcal{C}_L^{a,k}$ denotes a-times differentiability with Lipschitz k-th order derivatives. "SC" means "strongly-convex". $\widetilde{O}(\cdot)$ compresses logarithmic factors of $1/\epsilon$ and $1/\delta$, where $\delta \in (0,1)$ denotes the failure probability.

Method [3]	Loop Style	Stochastic Setting	Oracle Complexity	Upper-Level f	Lower-Level g	Batch Size
BSA (Ghadimi & Wang, 2018)	Double	General expectation	$\widetilde{O}(\epsilon^{-6})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$\widetilde{O}(1)$
StocBio (Ji et al., 2021)	Double	General expectation	$\widetilde{O}(\epsilon^{-4})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$\widetilde{O}(\epsilon^{-2})$
AmIGO (Arbel & Mairal, 2021)	Double	General expectation	$\widetilde{O}(\epsilon^{-4})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$O(\epsilon^{-2})$
ALSET (Chen et al., 2021)	Double / Single [4]	General expectation	$O(\epsilon^{-4})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$O(1)$
TTSA (Hong et al., 2023)	Single	General expectation	$\widetilde{O}(\epsilon^{-5})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$\widetilde{O}(1)$
F^{2}SA (Kwon et al., 2023b)	Single	General expectation	$O(\epsilon^{-7})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$O(1)$
SOBA (Dagréou et al., 2022)	Single	Finite sum	$O(\epsilon^{-4})$	$\mathcal{C}_L^{2,2}$	SC and $\mathcal{C}_L^{3,3}$	$O(1)$
SABA (Dagréou et al., 2022)	Single	Finite sum	$O(N^{4/3}\epsilon^{-2})$ [5]	$\mathcal{C}_L^{2,2}$	SC and $\mathcal{C}_L^{3,3}$	$O(1)$
MA-SOBA (Chen et al., 2023b)	Single	General expectation	$O(\epsilon^{-4})$	$\mathcal{C}_L^{1,1}$	SC and $\mathcal{C}_L^{2,2}$	$O(1)$
BO-REP (Hao et al., 2024)	Double	General expectation	$\widetilde{O}(\epsilon^{-4})$	$(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smooth	SC and $\mathcal{C}_L^{2,2}$	$O(1)$
SLIP (This work, Theorem 4.1)	Single	General expectation	$\widetilde{O}(\epsilon^{-4})$	$(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smooth	SC and $\mathcal{C}_L^{2,2}$	$O(1)$
SLIP (This work, Theorem 4.3)	Single	High probability	$\widetilde{O}(\epsilon^{-4})$	$(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smooth	SC and $\mathcal{C}_L^{2,2}$	$O(1)$

B. Properties of $(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-Smoothness Assumption (Assumption 3.1)

B.1. Definitions of Relaxed Smoothness

The standard relaxed smoothness assumption originally introduced in (Zhang et al., 2020c) is defined in Definition B.1.

Definition B.1 ((Zhang et al., 2020c))**.** A twice differentiable function F is (K_0, K_1)-smooth if $\|\nabla^2 F(w)\| \leq K_0 + K_1\|\nabla F(w)\|$ for any w.

The following Definition B.2 is an alternative definition for the (K_0, K_1)-smoothness. It does not need f to be twice differentiable and is strictly weaker than L-smoothness.

Definition B.2 (Remark 2.3 in (Zhang et al., 2020a))**.** A differentiable function F is (K_0, K_1)-smooth if $\|\nabla F(w) - \nabla F(w')\| \leq (K_0 + K_1\|\nabla F(w)\|)\|w - w'\|$ for any $\|w - w'\| \leq 1/K_1$.

B.2. Relationships between Assumption 3.1 and Standard Relaxed Smoothness

The folowing lemma shows that $(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smoothness assumption (e.g., Assumption 3.1) can recover the standard relaxed smoothness (e.g., Definition B.2) when the upper-level variable x and the lower-level variable y have the

[3] We omit the comparison with variance reduction-based methods (except for SABA (Dagréou et al., 2022)) that may achieve $\widetilde{O}(\epsilon^{-3})$ complexity under additional mean-squared smoothness assumptions on both upper-level and lower-level problems, e.g., VRBO, MRBO (Yang et al., 2021); SUSTAIN (Khanduri et al., 2021); SVRB (Guo et al., 2021); or under the finite sum setting, e.g., SRBA (Dagréou et al., 2023).

[4] ALSET can converge without the need for double loops, but at the cost of a worse dependence on $\kappa := l_{g,1}/\mu$ in oracle complexity.

[5] SABA (Dagréou et al., 2022) studies finite-sum problem and adapts variance reduction technique SAGA (Defazio et al., 2014), here $N = m + n$ denotes the total number of samples.

same smoothness constants.

Lemma B.3 (Lemma 6 in (Hao et al., 2024)). *Let $L_{x,0} = L_{y,0} = K_0/2$ and $L_{x,1} = L_{y,1} = K_1/2$, then Assumption 3.1 implies that for any $w = (x, y), w' = (x', y')$ such that $\|w - w'\| \le 1/K_1$, we have*

$$\|\nabla_w f(w) - \nabla_w f(w')\| \le (K_0 + K_1\|\nabla_w f(w)\|)\|w - w'\|.$$

In other words, $(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smoothness assumption (Assumption 3.1) can recover the standard relaxed smoothness assumption (Definition B.2).

C. Auxiliary Lemmas for Bilevel Optimization Problems

Lemma C.1 (Hypergradient formula, Lemma 7 in (Hao et al., 2024)). *The hypergradient $\nabla\Phi(x)$ takes the forms of*

$$\begin{aligned}\nabla\Phi(x) &= \nabla_x f(x, y^*(x)) - \nabla^2_{xy} g(x, y^*(x))[\nabla^2_{yy} g(x, y^*(x))]^{-1}\nabla_y f(x, y^*(x)) \\ &= \nabla_x f(x, y^*(x)) - \nabla^2_{xy} g(x, y^*(x)) z^*(x),\end{aligned}$$

where $z^*(x)$ is the solution to the following linear system:

$$z^*(x) = \arg\min_z \frac{1}{2}\langle \nabla^2_{yy} g(x, y^*(x))z, z\rangle - \langle \nabla_y f(x, y^*(x)), z\rangle.$$

Lemma C.2 (Lipschitz property, Lemma 8 in (Hao et al., 2024)). *Suppose Assumptions 3.1 and 3.2 hold, then we have*

- *$y^*(x)$ is $(l_{g,1}/\mu)$-Lipschitz continuous.*
- *$z^*(x)$ is l_{z^*}-Lipschitz continuous, i.e.,*

$$\|z^*(x) - z^*(x')\| \le l_{z^*}\|x - x'\| \quad \text{if} \quad \|x - x'\| \le \frac{1}{\sqrt{2(1 + l^2_{g,1}/\mu^2)(L^2_{x,1} + L^2_{y,1})}},$$

where l_{z^*} is defined as

$$l_{z^*} := \sqrt{1 + \frac{l^2_{g,1}}{\mu^2}}\left(\frac{l_{g,2} l_{f,0}}{\mu^2} + \frac{1}{\mu}(L_{y,0} + L_{y,1} l_{f,0})\right).$$

Lemma C.3 ((L_0, L_1)-Smoothness, Lemma 9 in (Hao et al., 2024)). *Suppose Assumptions 3.1 and 3.2 hold. Then for any x, x' we have*

$$\|\nabla\Phi(x) - \nabla\Phi(x')\| \le (L_0 + L_1\|\nabla\Phi(x')\|)\|x - x'\| \quad \text{if} \quad \|x - x'\| \le \frac{1}{\sqrt{2(1 + l^2_{g,1}/\mu^2)(L^2_{x,1} + L^2_{y,1})}},$$

where (L_0, L_1)-smoothness constant L_0 and L_1 are defined as

$$L_0 = \sqrt{1 + \frac{l^2_{g,1}}{\mu^2}}\left(L_{x,0} + L_{x,1}\frac{l_{g,1} l_{f,0}}{\mu} + \frac{l_{g,1}}{\mu}(L_{y,0} + L_{y,1} l_{f,0}) + l_{f,0}\frac{l_{g,1} l_{g,2} + l_{g,2}\mu}{\mu^2}\right) \quad \text{and} \quad L_1 = \sqrt{1 + \frac{l^2_{g,1}}{\mu^2}} L_{x,1}.$$

Lemma C.4 (Descent inequality, Lemma 10 in (Hao et al., 2024)). *Suppose Assumptions 3.1 and 3.2 hold. Then for any x, x' we have*

$$\Phi(x) \le \Phi(x') + \langle \nabla\Phi(x'), x - x'\rangle + \frac{L_0 + L_1\|\nabla\Phi(x')\|}{2}\|x - x'\|^2 \quad \text{if} \quad \|x - x'\| \le \frac{1}{\sqrt{2(1 + l^2_{g,1}/\mu^2)(L^2_{x,1} + L^2_{y,1})}}.$$

D. Omitted Proofs in Section 4.2.1

D.1. Tracking the minimizer of lower-level function: high-probability guarantees

In this section we present a few useful lemmas in (Cutler et al., 2023). In particular, Lemma D.1 and D.2 serve as a starting point and provide standard one-step improvement guarantee and distance recursion formulation for tracking the minimizer of the lower-level objective function. Then one can apply Proposition D.3, which is a technical result for recursive control, to obtain the distance tracking result with high probability as shown in Lemma D.4. We present some of the proofs under notations in this paper as below for completeness.

Lemma D.1 (One-step improvement, Lemma 2 and 23 in (Cutler et al., 2023))**.** *Consider Algorithm 1 with arbitrary sequence $\{\tilde{x}_t\}$, for any $y \in \mathbb{R}^{d_y}$ and $t \geq 1$, we have the following estimate:*

$$2\alpha(g(\tilde{x}_t, \tilde{y}_{t+1}) - g(\tilde{x}_t, y)) \leq (1-\mu\alpha)\|\tilde{y}_t - y\|^2 - \|\tilde{y}_{t+1} - y\|^2 + 2\alpha\langle u_t, y - \tilde{y}_t\rangle + \frac{\alpha^2}{1 - l_{g,1}\alpha}\|u_t\|^2. \tag{D.1}$$

where we define $u_t = \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t) - \nabla_y g(\tilde{x}_t, \tilde{y}_t)$ to be the bias of the gradient estimator at time t.

Proof of Lemma D.1. Since $g(x, y)$ is $l_{g,1}$-smooth jointly in (x, y), we have

$$\begin{aligned} g(\tilde{x}_t, \tilde{y}_{t+1}) &\leq g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y g(\tilde{x}_t, \tilde{y}_t), \tilde{y}_{t+1} - \tilde{y}_t\rangle + \frac{l_{g,1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 \\ &= g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), \tilde{y}_{t+1} - \tilde{y}_t\rangle + \frac{l_{g,1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \langle u_t, \tilde{y}_t - \tilde{y}_{t+1}\rangle. \end{aligned}$$

For any given $\delta_t > 0$, applying Young's inequality yields

$$\langle u_t, \tilde{y}_t - \tilde{y}_{t+1}\rangle \leq \frac{\delta_t}{2}\|u_t\|^2 + \frac{1}{2\delta_t}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2.$$

Then for any given $y \in \mathbb{R}^{d_y}$, we have

$$\begin{aligned} g(\tilde{x}_t, \tilde{y}_{t+1}) &\leq g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), \tilde{y}_{t+1} - \tilde{y}_t\rangle + \frac{l_{g,1} + \delta_t^{-1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \frac{\delta_t}{2}\|u_t\|^2 \\ &= g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), \tilde{y}_{t+1} - \tilde{y}_t\rangle + \frac{1}{2\alpha}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \frac{l_{g,1} + \delta_t^{-1} - \alpha^{-1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \frac{\delta_t}{2}\|u_t\|^2 \\ &\leq g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), y - \tilde{y}_t\rangle + \frac{1}{2\alpha}\|y - \tilde{y}_t\|^2 - \frac{1}{2\alpha}\|y - \tilde{y}_{t+1}\|^2 + \frac{l_{g,1} + \delta_t^{-1} - \alpha^{-1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \frac{\delta_t}{2}\|u_t\|^2, \end{aligned}$$

where the last inequality holds since $\tilde{y}_{t+1} = \tilde{y}_t - \alpha\nabla_y g(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t)$ is the unique minimizer of the α^{-1}-strongly convex function $h(y) = \langle \nabla_y g(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), y - \tilde{y}_t\rangle + \frac{1}{2\alpha}\|y - \tilde{y}_t\|^2$, and thus $h(y) - h(\tilde{y}_{t+1}) \geq \frac{1}{2\alpha}\|y - \tilde{y}_{t+1}\|^2$ holds for any $y \in \mathbb{R}^{d_y}$. Now by μ-strong convexity of $g(x, y)$ in terms of y, we estimate

$$\begin{aligned} g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y G(\tilde{x}_t, \tilde{y}_t; \tilde{\pi}_t), y - \tilde{y}_t\rangle &= g(\tilde{x}_t, \tilde{y}_t) + \langle \nabla_y g(\tilde{x}_t, \tilde{y}_t), y - \tilde{y}_t\rangle + \langle u_t, y - \tilde{y}_t\rangle \\ &\leq g(\tilde{x}_t, y) - \frac{l_{g,1}}{2}\|y - \tilde{y}_t\|^2 + \langle u_t, y - \tilde{y}_t\rangle. \end{aligned}$$

Therefore we have

$$\begin{aligned} g(\tilde{x}_t, \tilde{y}_{t+1}) \leq g(\tilde{x}_t, y) &- \frac{l_{g,1}}{2}\|y - \tilde{y}_t\|^2 + \langle u_t, y - \tilde{y}_t\rangle + \frac{1}{2\alpha}\|y - \tilde{y}_t\|^2 - \frac{1}{2\alpha}\|y - \tilde{y}_{t+1}\|^2 \\ &+ \frac{l_{g,1} + \delta_t^{-1} - \alpha^{-1}}{2}\|\tilde{y}_{t+1} - \tilde{y}_t\|^2 + \frac{\delta_t}{2}\|u_t\|^2. \end{aligned}$$

Finally, taking $\delta_t = \alpha/(1 - l_{g,1}\alpha)$ and rearranging yields

$$2\alpha(g(\tilde{x}_t, \tilde{y}_{t+1}) - g(\tilde{x}_t, y)) \leq (1-\mu\alpha)\|\tilde{y}_t - y\|^2 - \|\tilde{y}_{t+1} - y\|^2 + 2\alpha\langle u_t, y - \tilde{y}_t\rangle + \frac{\alpha^2}{1 - l_{g,1}\alpha}\|u_t\|^2.$$

□

Lemma D.2 (Distance recursion, Lemma 25 in (Cutler et al., 2023)). *Consider Algorithm 1 with arbitrary sequence $\{\tilde{x}_t\}$, for any $t \geq 1$, we have the following recursion:*

$$\|\tilde{y}_{t+1} - \tilde{y}_{t+1}^*\|^2 \leq (1-\mu\alpha)\|\tilde{y}_t - \tilde{y}_t^*\|^2 + 2\alpha\langle u_t, \tilde{y}_t^* - \tilde{y}_t\rangle + \frac{\alpha^2}{1-l_{g,1}\alpha}\|u_t\|^2 + \left(1+\frac{1}{\mu\alpha}\right)D_t^2, \tag{D.2}$$

where we define $D_t \coloneqq \|\tilde{y}_t^* - \tilde{y}_{t+1}^*\|$ to be the minimizer drift at time t.

Proof of Lemma D.2. Note that the μ-strong convexity of $g(x,y)$ in terms of y implies

$$g(\tilde{x}_t, \tilde{y}_{t+1}) - g(\tilde{x}_t, \tilde{y}_t^*) \geq \frac{\mu}{2}\|\tilde{y}_{t+1} - \tilde{y}_t^*\|^2.$$

Combing this estimate with Lemma D.1 under the identification $y = \tilde{y}_t^*$ yields

$$(1+\mu\alpha)\|\tilde{y}_{t+1} - \tilde{y}_t^*\|^2 \leq (1-\mu\alpha)\|\tilde{y}_t - \tilde{y}_t^*\|^2 + 2\alpha\langle u_t, \tilde{y}_t^* - \tilde{y}_t\rangle + \frac{\alpha^2}{1-l_{g,1}\alpha}\|u_t\|^2.$$

Then we apply Young's inequality to obtain

$$\|\tilde{y}_{t+1} - \tilde{y}_{t+1}^*\|^2 \leq (1+\mu\alpha)\|\tilde{y}_{t+1} - \tilde{y}_t^*\|^2 + (1+(\mu\alpha)^{-1})\|\tilde{y}_t^* - \tilde{y}_{t+1}^*\|^2.$$

Combining the above inequalities yields

$$\begin{aligned}
\|\tilde{y}_{t+1} - \tilde{y}_{t+1}^*\|^2 &\leq (1+\mu\alpha)\|\tilde{y}_{t+1} - \tilde{y}_t^*\|^2 + (1+(\mu\alpha)^{-1})\|\tilde{y}_t^* - \tilde{y}_{t+1}^*\|^2 \\
&\leq (1-\mu\alpha)\|\tilde{y}_t - \tilde{y}_t^*\|^2 + 2\alpha\langle u_t, \tilde{y}_t^* - \tilde{y}_t\rangle + \frac{\alpha^2}{1-l_{g,1}\alpha}\|u_t\|^2 + (1+(\mu\alpha)^{-1})\|\tilde{y}_t^* - \tilde{y}_{t+1}^*\|^2 \\
&= (1-\mu\alpha)\|\tilde{y}_t - \tilde{y}_t^*\|^2 + 2\alpha\langle u_t, \tilde{y}_t^* - \tilde{y}_t\rangle + \frac{\alpha^2}{1-l_{g,1}\alpha}\|u_t\|^2 + \left(1+\frac{1}{\mu\alpha}\right)D_t^2,
\end{aligned}$$

where the last equality holds by definition of minimizer drift D_t. □

Proposition D.3 (Recursive control on MGF, Proposition 29 in (Cutler et al., 2023)). *Consider scalar stochastic processes (V_t), (U_t), and (X_t) on a probability space with filtration $(\mathcal{H}_t)$, which are linked by the inequality*

$$V_{t+1} \leq \alpha_t V_t + U_t\sqrt{V_t} + X_t + \kappa_t \tag{D.3}$$

for some deterministic constants $\alpha_t \in (-\infty, 1]$ and $\kappa_t \in \mathbb{R}$. Suppose the following properties hold.

- *V_t is non-negative and $\mathcal{H}_t$-measurable.*
- *U_t is mean-zero sub-Gaussian conditioned on $\mathcal{H}_t$ with deterministic parameter σ_t:*

$$\mathbb{E}[\exp(\lambda U_t) \mid \mathcal{H}_t] \leq \exp(\lambda^2\sigma_t^2/2), \quad \forall\lambda \in \mathbb{R}.$$

- *X_t is non-negative and sub-exponential conditioned on $\mathcal{H}_t$ with deterministic parameter ν_t:*

$$\mathbb{E}[\exp(\lambda U_t) \mid \mathcal{H}_t] \leq \exp(\lambda\nu_t), \quad \forall\lambda \in [0, 1/\nu_t].$$

Then the estimate

$$\mathbb{E}[\exp(\lambda V_{t+1})] \leq \exp(\lambda(\nu_t + \kappa_t))\mathbb{E}[\exp(\lambda(1+\alpha_t)V_t/2)]$$

holds for any λ satisfying $0 \leq \lambda \leq \min\left(\frac{1-\alpha_t}{2\sigma_t^2}, \frac{1}{2\nu_t}\right)$.

Lemma D.4 (High-probability distance tracking, Theorem 6 and 30 in (Cutler et al., 2023)). *Suppose that Assumption 3.2 holds and let $\{\tilde{y}_t\}$ be the iterates produced by Algorithm 1 with constant learning rate $\alpha \le 1/(2l_{g,1})$. We further assume there exists constant $D > 0$ such that the drift D_t^2 is sub-exponential conditioned on $\mathcal{F}_t^1$ with parameter D^2:*

$$\mathbb{E}\left[\exp(\lambda D_t^2) \mid \mathcal{F}_t^1\right] \le \exp(\lambda D^2) \quad \textit{for all} \quad 0 \le \lambda \le D^{-2}.$$

Then for any fixed $t \in [T]$ and $\delta \in (0,1)$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_t^1$:

$$\|\tilde{y}_t - \tilde{y}_t^*\|^2 \le \left(1 - \frac{\mu\alpha}{2}\right)^t \|\tilde{y}_0 - \tilde{y}_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4D^2}{\mu^2\alpha^2}\right)\log\left(\frac{e}{\delta}\right), \tag{D.4}$$

where e denotes the base of natural logarithms.

Remark: We would like to claim that in original bound for (D.4), there is an extra c^2 on the variance term, namely $c^2\sigma_{g,1}^2$ instead of c^2. By statement of Theorem 30 (footnote 6) in (Cutler et al., 2023), the absolute constant satisfies $c \ge 1$, so here in (D.4) we just take $c = 1$ for simplicity.

D.2. Proof of Lemma 4.4

In this section we establish the tracking error of the lower-level problem $\|y_t - y_t^*\|$ at each iteration for *any fixed slowly changing upper-level sequence* $\{\tilde{x}_t\}$ with high probability. Our result (Lemma 4.4) can be seen as a direct generalization of Lemma D.4. It will be applied to two stages of Algorithm 2, which helps us build a refined characterization of the lower-level problem. To be more specific, Lemma 4.5 corresponds to warm-start stage in line 3, and Lemma 4.6 corresponds to simultaneous update stage in line 4∼10.

Lemma D.5 (Restatement of Lemma 4.4). *Suppose that Assumption 3.2 holds, let $\{\tilde{y}_t\}$ be the iterates produced by Algorithm 1 with any fixed input sequence $\{\tilde{x}_t\}$ such that $\|\tilde{x}_{t+1} - \tilde{x}_t\| \le R$ for all $t \ge 0$, and constant learning rate $\alpha \le 1/(2l_{g,1})$. Then for any fixed $t \in [N]$ and $\delta \in (0,1)$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_t^1$:*

$$\|\tilde{y}_t - \tilde{y}_t^*\|^2 \le \left(1 - \frac{\mu\alpha}{2}\right)^t \|\tilde{y}_0 - \tilde{y}_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4R^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{e}{\delta}\right). \tag{D.5}$$

As a consequence, for any given $\delta \in (0,1)$ and all $t \in [N]$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_T^1$:

$$\|\tilde{y}_t - \tilde{y}_t^*\|^2 \le \left(1 - \frac{\mu\alpha}{2}\right)^t \|\tilde{y}_0 - \tilde{y}_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4R^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eN}{\delta}\right). \tag{D.6}$$

Proof of Lemma D.5. By Lemma C.2, $y^*(x)$ is $(l_{g,1}/\mu)$-Lipschitz continuous, then $D_t = \|\tilde{y}_t^* - \tilde{y}_{t+1}^*\| \le \frac{l_{g,1}}{\mu}\|\tilde{x}_t - \tilde{x}_{t+1}\| = l_{g,1}R/\mu$ holds for any $t \in [N]$ almost surely, hence we can choose $D = l_{g,1}R/\mu$ by Lemma D.4. Replacing D^2 with $l_{g,1}^2R^2/\mu^2$ in (D.4) yields (D.5), and we obtain (D.6) by using union bound. □

D.3. Proof of Lemma 4.5

In this section we apply Lemma 4.4 to obtain constant error of the lower-level variable (i.e., $\|y_0 - y_0^*\|$) with high probability within a logarithmic number of iterations.

Lemma D.6 (Warm-start, Restatement of Lemma 4.5). *Suppose that Assumption 3.2 holds and given any $\delta \in (0,1)$, let $\{y_t^{\mathsf{init}}\}$ be the iterates produced by Algorithm 1 starting from y_0^{init} with $R = 0$ (where R is defined in Lemma 4.4, it means that $\tilde{x}_t = x_0$ for any t) and constant learning rate α^{init} satisfying*

$$\alpha^{\mathsf{init}} = \min\left\{\frac{1}{2l_{g,1}}, \frac{\mu}{2048L_1^2\sigma_{g,1}^2\log(e/\delta)}\right\} = O(1). \tag{D.7}$$

Then Algorithm 1 guarantees $\|y_{T_0}^{\mathsf{init}} - y_0^*\| \le 1/(8\sqrt{2}L_1)$ with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_{T_0}^1$ (we denote this event as $\mathcal{E}_{\mathsf{init}}$), where the number of iterations T_0 satisfies

$$T_0 = \frac{\log\left(256L_1^2\|y_0^{\mathsf{init}} - y_0^*\|^2\right)}{\log\left(2/(2-\mu\alpha^{\mathsf{init}})\right)} = \widetilde{O}(1). \tag{D.8}$$

Proof of Lemma D.6. We set $R = 0$ in (D.5) such that $\tilde{x}_t = x_0$ for any $t \in [T_0]$, and learning rate satisfies $\alpha \le 1/(2l_{g,1})$ by (D.7). Then by Lemma D.5, for any $\delta \in (0,1)$, with probability at least $1-\delta$ over the randomness in $\widetilde{\mathcal{F}}_{T_0}^1$ we have

$$\begin{aligned}\|y_{T_0}^{\mathsf{init}} - y_0^*\|^2 &\le \left(1 - \frac{\mu\alpha^{\mathsf{init}}}{2}\right)^{T_0}\|y_0^{\mathsf{init}} - y_0^*\|^2 + \frac{8\alpha^{\mathsf{init}}\sigma_{g,1}^2}{\mu}\log\left(\frac{e}{\delta}\right)\\ &\le \frac{1}{256L_1^2} + \frac{1}{256L_1^2} = \frac{1}{128L_1^2},\end{aligned}$$

where we use (D.7) and (D.8) for the last inequality. Therefore, we obtain $\|y_{T_0}^{\mathsf{init}} - y_0^*\| \le 1/(8\sqrt{2}L_1)$ under event $\mathcal{E}_{\mathsf{init}}$. □

D.4. Proof of Lemma 4.6

In this section we again apply Lemma 4.4 with proper parameter setting to obtain a refined control of the lower-level variable.

Lemma D.7 (Restatement of Lemma 4.6)**.** *Under Assumptions 3.1, 3.2 and event $\mathcal{E}_{\mathsf{init}}$, for any given $\delta \in (0,1)$ and any small ϵ satisfying*

$$\begin{aligned}\epsilon \le \min\Bigg\{&\frac{L_0}{L_1}, \Delta_{y,0}L_0, \frac{8l_{g,1}L_0}{\mu\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}, \sqrt{\frac{16el_{g,1}\Delta_0 L_0}{\mu\delta}}, 4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4};\\ &\sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu/(2l_{g,1}))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu l_{g,1}\Delta_{z,0}/(2\Delta_0 L_0))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}},\\ &\frac{\Delta_0 L_0}{\|\nabla\Phi(x_0)\|}, \frac{L_0\sigma_{g,1}}{\sigma_{g,2}}, \frac{L_0\sigma_{g,1}}{\sqrt{\mu l_{g,1}}}, \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}\exp\left(\frac{-l_{g,1}\sqrt{\sigma_{f,1}^2 + 2l_{f,0}^2\sigma_{g,2}^2/\mu^2}}{512L_0\sigma_{g,1}}\right)\Bigg\},\end{aligned} \tag{D.9}$$

if we choose parameters $\alpha, \beta, \gamma, \eta$ as

$$\begin{aligned}1-\beta = \min\Bigg\{&1, \frac{\mu}{16l_{g,1}}, \frac{16el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2}, \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}; \frac{\min\{1,\mu^2/(32l_{g,1}^2)\}}{\sigma_{f,1}^2 + 2l_{f,0}^2\sigma_{g,2}^2/\mu^2}\epsilon^2, \frac{l_{g,1}^2}{8\sigma_{g,2}^2}, \frac{\mu^2}{16\sigma_{g,2}^2};\\ &\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu/(2l_{g,1}))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu l_{g,1}\Delta_{z,0}^2/(2\Delta_0 L_0))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}\Bigg\},\end{aligned} \tag{D.10}$$

$$\eta = \min\left\{\frac{1}{8}\min\left(\frac{1}{L_1}, \frac{\epsilon}{L_0}, \frac{\epsilon\Delta_0}{\Delta_{y,0}^2 L_0^2}, \frac{\Delta_0}{\|\nabla\Phi(x_0)\|}, \frac{\epsilon\Delta_0}{l_{g,1}^2\Delta_{z,0}^2}, \frac{\mu\epsilon}{l_{g,1}L_0\log(A)}\right)(1-\beta), \frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}\right\}, \tag{D.11}$$

$$\gamma = \frac{1}{\mu}(1-\beta), \qquad \alpha = \frac{8}{\mu}(1-\beta), \tag{D.12}$$

where e denotes the base of natural logarithms, and we define $\Delta_0, \Delta_{y,0}, \Delta_{z,0}, A$ and B as

$$\Delta_0 := \Phi(x_0) - \inf_{x\in\mathbb{R}^{d_x}}\Phi(x), \quad \Delta_{y,0} := \|y_0 - y_0^*\|, \quad \Delta_{z,0} := \|z_0 - z_0^*\|, \quad A := \left(\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2(1-\beta)}\right)^2, \quad B := \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta\epsilon^4}\right)^4. \tag{D.13}$$

Run Algorithm 2 for $T = \frac{4\Delta_0}{\eta\epsilon}$ iterations, then for all $t \in [T]$, Algorithm 2 guarantees with probability at least $1-\delta$ over the randomness in $\mathcal{F}_T^1$ (we denote this event as $\mathcal{E}_y$) that:

1. $\|y_t - y_t^*\| \le \frac{1}{8L_1}$.
2. $\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|y_i - y_i^*\| \le \frac{3}{32L_0}\epsilon$.

Proof of Lemma D.7. To begin, by choice of β and η as in (D.10) and (D.11) we have

$$\alpha = 8\gamma = \frac{8(1-\beta)}{\mu} \le \frac{8}{\mu}\frac{\mu}{16l_{g,1}} \le \frac{1}{2l_{g,1}} \qquad \text{and} \qquad \eta \le \frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}},$$

thus satisfy the condition for applying Lemma D.5 and Lemma C.3, i.e., (L_0, L_1)-smoothness of function Φ. By Lemma D.5 (we set $R = \eta$ here) and choice of α, γ, T as in (D.12), with probability at least $1-\delta$ over the randomness in $\mathcal{F}_t^1$ (we denote this event as $\mathcal{E}_y$) we have

$$\begin{aligned}\|y_t - y_t^*\|^2 &\le \left(1-\frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right) \\ &= \left(1-\frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 + \left(\frac{64\gamma\sigma_{g,1}^2}{\mu} + \frac{\eta^2 l_{g,1}^2}{16\mu^4\gamma^2}\right)\log\left(\frac{eT}{\delta}\right) \\ &= \left(1-\frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 + \left(\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2}\right)\log\left(\frac{4e\Delta_0}{\eta\delta\epsilon}\right).\end{aligned} \tag{D.14}$$

Before delving into details, let's first briefly outline the structure of the proof for this lemma.

Proof for $\|y_t - y_t^*\| \le \frac{1}{8L_1}$:

For the first part of the proof, we split it into the following four parts.

In **step 1**, we claim that with choice of ϵ and β as in (D.9) and (D.10), the exact formula for η and $1-\beta$ are as the following:

$$\eta = \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta) \qquad \text{and} \qquad 1-\beta = \frac{\mu^2\epsilon^2}{64\cdot 1024 L_0^2\sigma_{g,1}^2\log^2(B)}. \tag{D.15}$$

In **step 2**, we show under event $\mathcal{E}_{\mathsf{init}}$, for any $t \in [T]$, it holds that

$$\left(1-\frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 \le \frac{1}{128L_1^2}.$$

In **step 3**, we show that with suitable choice of ϵ, we have

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} \le \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2},$$

and thus we could combine the above two terms together

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} \le \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} = \frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}.$$

In **step 4**, we show that

$$\frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}\log\left(\frac{4e\Delta_0}{\eta\delta\epsilon}\right) \le \frac{\epsilon^2}{128L_0^2}.$$

Finally we merge the above four steps together and obtain

$$\|y_t - y_t^*\|^2 \le \frac{1}{128L_1^2} + \frac{\epsilon^2}{512L_0^2}$$

holds for any $t \in [T]$, and again by choice of $\epsilon \le L_0/L_1$ as in (D.9) we complete the first part of the proof. Now we begin proof step by step accordingly.

Step 1. One could check Lemma D.14 for details.

Step 2. By Lemma D.6, under event $\mathcal{E}_{\mathsf{init}}$, for any $t \in [T]$ we have

$$\left(1-\frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 \le \|y_0 - y_0^*\|^2 = \|y_{T_0}^{\mathsf{init}} - y_0^*\|^2 \le \frac{1}{128L_1^2}.$$

Step 3. With choice of η as in (D.15), it suffices to show that

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} \le \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} \quad \Longleftrightarrow \quad \frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} \le \frac{\epsilon^2}{1024L_0^2\log^2(A)},$$

which is equivalent to

$$(1-\beta)\log^2(A) \le \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2} \quad \Longleftrightarrow \quad \log^2(A) \le \log^2(B) \quad \Longleftrightarrow \quad \log(A) \le \log(B),$$

where we use $A \ge 4$ and $B \ge 4$ by choice of β and ϵ as in (D.10) and (D.9), i.e.,

$$1-\beta \le \frac{16el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2}, \quad \epsilon \le 4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}.$$

Plugging in the definition of A and B leads to

$$\log(A) \le \log(B) \quad \Longleftrightarrow \quad A \le B \quad \Longleftrightarrow \quad \left(\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2(1-\beta)}\right)^2 \le \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta\epsilon^4}\right)^4. \tag{D.16}$$

Recall (D.15), step 1 claims that

$$1-\beta = \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}.$$

Plug the above expression of $1-\beta$ into (D.16), then (D.16) turns into the inequality $\sqrt{B}\log^4(B) \le B$. To summarize, now we have the following equivalence:

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} \le \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} \quad \Longleftrightarrow \quad \sqrt{B}\log^4(B) \le B.$$

Thus we only need to set $B \ge 22\times 10^{10}$ to satisfy the above inequality in order to make the claim of step 3 hold. In fact, with choice of ϵ as in (D.9), one can easily verify that

$$\epsilon \le 4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4} \quad \Longrightarrow \quad B = \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta\epsilon^4}\right)^4 \ge 22\times 10^{10}.$$

This completes the proof of step 3 and we conclude that

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} \le \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2} = \frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}.$$

Step 4. Under choice of η and β as in (D.15) and (D.10), we have $A \ge 4$ and thus

$$\frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}\log\left(\frac{4e\Delta_0}{\eta\delta\epsilon}\right) = \frac{\epsilon^2}{512L_0^2\log^2(A)}\log\left(\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2(1-\beta)}\log(A)\right) \le \frac{\epsilon^2}{512L_0^2\log(A)}\log\left(\sqrt{A}\log(A)\right) \le \frac{\epsilon^2}{512L_0^2},$$

where we use the fact that $\log(\sqrt{A}\log(A)) \le \log(A) \le \log^2(A)$ for any $A \ge 4$.

Merge Step. Finally, merging the above four steps and combining with (D.14) gives that, under event $\mathcal{E}_{\mathsf{init}}$, for any $t \in [T]$,

$$\begin{aligned}
\|y_t - y_t^*\|^2 &\le \left(1-\frac{\mu\alpha}{2}\right)^t\|y_0-y_0^*\|^2 + \left(\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2} + \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2}\right)\log\left(\frac{4e\Delta_0}{\eta\delta\epsilon}\right) \\
&\le \left(1-\frac{\mu\alpha}{2}\right)^t\|y_0-y_0^*\|^2 + \frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}\log\left(\frac{4e\Delta_0}{\eta\delta\epsilon}\right) \\
&\le \frac{1}{128L_1^2} + \frac{\epsilon^2}{512L_0^2},
\end{aligned} \tag{D.17}$$

which, together with $\epsilon \le L_0/L_1$ yields the first part of the result.

Proof for $(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|y_i-y_i^*\|\leq\frac{3\epsilon}{32L_0}$:

As for the second part, under event $\mathcal{E}_{\text{init}}\cap\mathcal{E}_y$ we have

$$\begin{aligned}
\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|y_i-y_i^*\| &\overset{(i)}{\leq} \frac{(1-\beta)}{T}\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\sqrt{\left(1-\frac{\mu\alpha}{2}\right)^i\|y_0-y_0^*\|^2+\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\\
&\overset{(ii)}{\leq} \frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\left[\left(1-\frac{\mu\alpha}{2}\right)^{i/2}\|y_0-y_0^*\|+\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right]\\
&\leq \frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\left[\beta^t\sum_{i=0}^{t}\left(\frac{\sqrt{1-\mu\alpha/2}}{\beta}\right)^i\|y_0-y_0^*\|+\frac{1}{1-\beta}\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right]\\
&\overset{(iii)}{\leq} \frac{4\Delta_{y,0}}{T(\mu\alpha-4(1-\beta))}+\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)},
\end{aligned}\tag{D.18}$$

where (i) follows by the first line of (D.14), (ii) follows by $\sqrt{a+b}\leq\sqrt{a}+\sqrt{b}$ for $a,b\geq 0$, (iii) follows since we have

$$\begin{aligned}
(1-\beta)\sum_{t=0}^{T-1}\beta^t\sum_{i=0}^{t}\left(\frac{\sqrt{1-\mu\alpha/2}}{\beta}\right)^i &\leq (1-\beta)\sum_{t=0}^{T-1}\beta^t\frac{\beta}{\beta-\sqrt{1-\mu\alpha/2}}\leq\frac{\beta}{\beta-\sqrt{1-\mu\alpha/2}}\leq\frac{\beta\left(\beta+\sqrt{1-\mu\alpha/2}\right)}{\mu\alpha/2-(1-\beta^2)}\\
&\leq\frac{2}{\mu\alpha/2-(1-\beta)(1+\beta)}\leq\frac{2}{\mu\alpha/2-2(1-\beta)}=\frac{4}{\mu\alpha-4(1-\beta)},
\end{aligned}$$

and also by definition of $\Delta_{y,0}$ in (D.13) and $\mu\alpha=8(1-\beta)$ by (D.12) and hence $\mu\alpha-4(1-\beta)>0$.

Moreover, we have

$$\begin{aligned}
\frac{4\Delta_{y,0}}{T(\mu\alpha-4(1-\beta))}+\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)} &\overset{(i)}{\leq}\frac{\Delta_{y,0}}{T(1-\beta)}+\sqrt{\frac{\epsilon^2}{512L_0^2}}\overset{(ii)}{=}\frac{\eta\epsilon\Delta_{y,0}}{4\Delta_0(1-\beta)}+\frac{\epsilon}{16\sqrt{2}L_0}\\
&\overset{(iii)}{\leq}\frac{\epsilon}{32L_0}+\frac{\epsilon}{16\sqrt{2}L_0}\leq\frac{3}{32L_0}\epsilon,
\end{aligned}\tag{D.19}$$

where (i) follows from $\mu\alpha=8(1-\beta)$, (D.14) and (D.17), (ii) follows from the choice of $T=4\Delta_0/(\eta\epsilon)$ and (iii) follows from $\eta\leq\epsilon\Delta_0(1-\beta)/(8\Delta_{y,0}^2L_0^2)\leq\Delta_0(1-\beta)/(8\Delta_{y,0}L_0)$ by $\epsilon\leq\Delta_{y,0}L_0$ from (D.37).

Combining (D.18) and (D.19) finally yields $(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|y_i-y_i^*\|\leq\frac{3\epsilon}{32L_0}$. □

D.5. Proof of Lemma 4.7

In this section, we aim to leverage the cumulative moving average hypergradient estimation error, namely Lemma 4.7. To this end, we present Lemma D.8, D.9 and D.10 to give upper bound for the cumulative error of the linear system estimator, the variance as well as the bias of the hypergradient estimator, respectively. It's worth noting that we can use Lemma 4.6 and independence of filtration to handle the most difficult part in Lemma D.10, namely $L_{x,1}\|y_t-y_t^*\|\|\nabla\Phi(x_t)\|$.

Lemma D.8. *Under Assumptions 3.1, 3.2 and event $\mathcal{E}_{\text{init}}\cap\mathcal{E}_y$, if we choose $\gamma\leq\min\{1/(4\mu),\mu/(16\sigma_{g,2}^2),\alpha/8\}$, then we have the following estimate:*

$$\begin{aligned}
\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2] &\leq\frac{1}{\mu\gamma}\|z_0-z_0^*\|^2+\frac{10(1-\mu\gamma)}{\mu^3(\alpha-2\gamma)}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\|y_0-y_0^*\|^2\\
&+T\left\{\frac{2\gamma}{\mu}\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2+\sigma_{f,1}^2\right)+\frac{4l_{z^*}^2}{\mu^2}\frac{\eta^2}{\gamma^2}+\frac{5}{\mu^2}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)\right\},
\end{aligned}\tag{D.20}$$

where the expectation is taken over the randomness in $\widetilde{\mathcal{F}}_T$.

Proof of Lemma D.8. Follow the same procedure as Lemma 13 in (Hao et al., 2024), we have

$$\mathbb{E}_t[\|z_{t+1}-z_{t+1}^*\|^2]\leq(1-\mu\gamma)\|z_t-z_t^*\|^2+\frac{5\gamma}{\mu}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\|y_t-y_t^*\|^2+2\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2+\sigma_{f,1}^2\right)\gamma^2+\frac{4}{\mu\gamma}l_{z^*}^2\eta^2.$$

Under event $\mathcal{E}_{\text{init}} \cap \mathcal{E}_y$, we have

$$\|y_t - y_t^*\|^2 \leq \left(1 - \frac{\mu\alpha}{2}\right)^t \|y_0 - y_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right) \log\left(\frac{eT}{\delta}\right),$$

which gives

$$\begin{aligned}
\mathbb{E}[\|z_t - z_t^*\|^2] \leq (1-\mu\gamma)^t \|z_0 - z_0^*\|^2 + \sum_{i=0}^{t-1} (1-\mu\gamma)^{t-i-1} \Bigg\{ & 2\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2 + \sigma_{f,1}^2\right)\gamma^2 + \frac{4}{\mu\gamma} l_{z^*}^2 \eta^2 \\
& + \frac{5\gamma}{\mu}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right) \left[\left(1 - \frac{\mu\alpha}{2}\right)^i \|y_0 - y_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right) \log\left(\frac{eT}{\delta}\right)\right] \Bigg\}.
\end{aligned}$$

Taking summation yields

$$\begin{aligned}
\sum_{t=0}^{T-1} \mathbb{E}[\|z_t - z_t^*\|^2] &\leq \sum_{t=0}^{T-1} (1-\mu\gamma)^t \|z_0 - z_0^*\|^2 + \sum_{t=0}^{T-1}\sum_{i=0}^{t-1} (1-\mu\gamma)^{t-i-1} \Bigg\{ 2\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2 + \sigma_{f,1}^2\right)\gamma^2 + \frac{4}{\mu\gamma} l_{z^*}^2 \eta^2 \\
&\quad + \frac{5\gamma}{\mu}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right) \left[\left(1 - \frac{\mu\alpha}{2}\right)^i \|y_0 - y_0^*\|^2 + \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right) \log\left(\frac{eT}{\delta}\right)\right] \Bigg\} \\
&\leq \sum_{t=0}^{T-1} (1-\mu\gamma)^t \|z_0 - z_0^*\|^2 + \sum_{t=0}^{T-1}\sum_{i=0}^{t-1} (1-\mu\gamma)^{t-i-1} \Bigg\{ 2\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2 + \sigma_{f,1}^2\right)\gamma^2 + \frac{4}{\mu\gamma} l_{z^*}^2 \eta^2 \\
&\quad + \frac{5\gamma}{\mu}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right) \left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right) \log\left(\frac{eT}{\delta}\right) \Bigg\} \\
&\quad + \frac{5\gamma}{\mu}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right) \sum_{t=0}^{T-1}\sum_{i=0}^{t-1} (1-\mu\gamma)^{t-i-1} \left(1 - \frac{\mu\alpha}{2}\right)^i \|y_0 - y_0^*\|^2.
\end{aligned}$$

Since $\gamma \leq \alpha/8$, then $\alpha - 2\gamma > 0$, for any $t_0 \in [T]$ we have

$$\sum_{t=0}^{T-1}\sum_{i=0}^{t-1} (1-\mu\gamma)^{t-i-1} \left(1 - \frac{\mu\alpha}{2}\right)^i = \sum_{t=0}^{T-1} (1-\mu\gamma)^{t-1} \sum_{i=0}^{t-1} \left(\frac{1 - \mu\alpha/2}{1-\mu\gamma}\right)^i \leq \sum_{t=0}^{T-1} (1-\mu\gamma)^{t-1} \frac{2(1-\mu\gamma)}{\mu(\alpha - 2\gamma)} \leq \frac{2(1-\mu\gamma)}{\mu^2\gamma(\alpha-2\gamma)}.$$

Therefore, we conclude that

$$\begin{aligned}
\sum_{t=0}^{T-1} \mathbb{E}[\|z_t - z_t^*\|^2] &\leq \frac{1}{\mu\gamma}\|z_0 - z_0^*\|^2 + \frac{10(1-\mu\gamma)}{\mu^3(\alpha-2\gamma)}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right)\|y_0 - y_0^*\|^2 \\
&\quad + T\left\{\frac{2\gamma}{\mu}\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2 + \sigma_{f,1}^2\right) + \frac{4l_{z^*}^2}{\mu^2}\frac{\eta^2}{\gamma^2} + \frac{5}{\mu^2}\left(\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} + (L_{y,0} + L_{y,1} l_{f,0})^2\right)\left(\frac{8\alpha\sigma_{g,1}^2}{\mu} + \frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)\right\}.
\end{aligned}$$

□

Lemma D.9 (Variance, Lemma 14 in (Hao et al., 2024)). *Under Assumptions 3.1, 3.2, we have*

$$\mathbb{E}[\|\widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t') - \mathbb{E}_t[\widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t')]\|^2] \leq \sigma_{f,1}^2 + \frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2 + 2\sigma_{g,2}^2 \mathbb{E}[\|z_t - z_t^*\|^2], \tag{D.21}$$

where we define $\widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t') = \nabla_x F(x_t, y_t; \xi_t') - \nabla_{xy}^2 G(x_t, y_t; \zeta_t') z_k$ as the hypergradient estimator, and the total expectation is taken over the randomness in $\widetilde{\mathcal{F}}_T$.

Lemma D.10 (Bias, Lemma 4 in (Hao et al., 2024)). *Under Assumptions 3.1, 3.2, we have*

$$\|\mathbb{E}_t[\widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t')] - \nabla\Phi(x_t)\| \leq L_{x,1}\|y_t - y_t^*\|\|\nabla\Phi(x_t)\| + \left(L_{x,0} + L_{x,1}\frac{l_{g,1} l_{f,0}}{\mu} + \frac{l_{g,2} l_{f,0}}{\mu}\right)\|y_t - y_t^*\| + l_{g,1}\|z_t - z_t^*\|. \tag{D.22}$$

With Lemma D.8, D.9 and D.10, we are now ready to leverage the cumulative estimation error for the hypergradient.

Lemma D.11 (Restatement of Lemma 4.7)**.** *Under Assumptions 3.1, 3.2 and event $\mathcal{E}_{\text{init}} \cap \mathcal{E}_y$, define $\epsilon_t = m_{t+1} - \nabla\Phi(x_t)$ to be the moving-average hypergradient estimation error. Then we have*

$$\begin{aligned}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right] \le& \left(\frac{\eta L_1\beta}{1-\beta}+\frac{1}{8}\right)\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| + T\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2+\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2}+\frac{T\eta L_0\beta}{1-\beta}\\ &+\sqrt{T}\left(\sqrt{2}\sigma_{g,2}\sqrt{1-\beta}+l_{g,1}\right)\sqrt{\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]}+\frac{\beta}{1-\beta}\|m_0-\nabla\Phi(x_0)\| \\ &+\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\left[\frac{4\Delta_{y,0}}{\mu\alpha-4(1-\beta)}+T\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right],\end{aligned} \tag{D.23}$$

where $\Delta_{y,0}$ is defined in (D.13), and the expectation is taken over the randomness in $\widetilde{\mathcal{F}}_T$.

Proof of Lemma D.11. We define $\epsilon_t = m_{t+1} - \nabla\Phi(x_t)$, $\hat{\epsilon}_t = \widehat{\nabla}\Phi(x_t, y_t, z_t; \xi_t', \zeta_t') - \nabla\Phi(x_t)$ and $S(a,b) = \nabla\Phi(a) - \nabla\Phi(b)$. It is easy to see that $\|S(a,b)\| \le (L_0 + L_1\|\nabla\Phi(a)\|)\|a-b\|$ by (L_0, L_1)-smoothness of function Φ. Specifically, $\|x_t - x_{t+1}\| = \eta$ holds true for any t and thus $\|S(x_t, x_{t+1})\| \le (L_0 + L_1\|\nabla\Phi(x_t)\|)\eta$. By definition of $\epsilon_t, \hat{\epsilon}_t$ and $S(a,b)$, we have the following recursion:

$$\epsilon_{t+1} = \beta\epsilon_t + \beta S(x_t, x_{t+1}) + (1-\beta)\hat{\epsilon}_{t+1}. \tag{D.24}$$

Then we apply (D.24) recursively and obtain

$$\epsilon_t = \beta^{t+1}(m_0 - \nabla\Phi(x_0)) + \beta\sum_{i=0}^{t-1}\beta^{t-i-1}S(x_i, x_{i+1}) + (1-\beta)\sum_{i=0}^{t}\beta^{t-i}\hat{\epsilon}_i,$$

which gives

$$\|\epsilon_t\| \le \beta^{t+1}\|m_0 - \nabla\Phi(x_0)\| + \eta\beta\sum_{i=0}^{t-1}\beta^{t-i-1}(L_0 + L_1\|\nabla\Phi(x_i)\|) + (1-\beta)\left\|\sum_{i=0}^{t}\beta^{t-i}\hat{\epsilon}_i\right\|. \tag{D.25}$$

Taking summation and total expectation yields

$$\begin{aligned}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right] &\le (1-\beta)\sum_{t=0}^{T-1}\mathbb{E}\left\|\sum_{i=0}^{t}\beta^{t-i}\hat{\epsilon}_i\right\| + \frac{\eta L_1\beta}{1-\beta}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| + \frac{T\eta L_0\beta}{1-\beta} + \frac{\beta}{1-\beta}\|m_0 - \nabla\Phi(x_0)\| \\ &\le (1-\beta)\sum_{t=0}^{T-1}\mathbb{E}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i') - \mathbb{E}_i[\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i')]\right)\right\| + \frac{\eta L_1\beta}{1-\beta}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \qquad \text{(D.26)} \\ &\quad + (1-\beta)\sum_{t=0}^{T-1}\mathbb{E}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\mathbb{E}_i[\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i')] - \nabla\Phi(x_i)\right)\right\| + \frac{T\eta L_0\beta}{1-\beta} + \frac{\beta}{1-\beta}\|m_0 - \nabla\Phi(x_0)\|. \qquad \text{(D.27)}\end{aligned}$$

For the first term of (D.26), we follow the same procedure as equation (68) of Lemma 5 in (Hao et al., 2024) and obtain

$$\begin{aligned}&(1-\beta)\sum_{t=0}^{T-1}\mathbb{E}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i') - \mathbb{E}_i[\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i')]\right)\right\| \\ &\qquad \le T\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2 + \frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2} + \sqrt{2}\sigma_{g,2}\sqrt{T}\sqrt{1-\beta}\sqrt{\sum_{t=0}^{T-1}\mathbb{E}[\|z_t - z_t^*\|^2]}.\end{aligned} \tag{D.28}$$

For the first term of (D.27), by triangle inequality and Lemma D.10 we have

$$\begin{aligned}&(1-\beta)\sum_{t=0}^{T-1}\mathbb{E}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\mathbb{E}_i[\widehat{\nabla}\Phi(x_i, y_i, z_i; \xi_i', \zeta_i')] - \nabla\Phi(x_i)\right)\right\| \le (1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}l_{g,1}\mathbb{E}[\|z_i - z_i^*\|] \\ &+ (1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}L_{x,1}\|y_i - y_i^*\|\mathbb{E}\|\nabla\Phi(x_i)\| + (1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\left(L_{x,0} + L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu} + \frac{l_{g,2}l_{f,0}}{\mu}\right)\|y_i - y_i^*\|.\end{aligned} \tag{D.29}$$

Now we proceed to upper bound the right-hand side of (D.29), respectively.

For the first term on right-hand side of (D.29), we have

$$(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}l_{g,1}\mathbb{E}[\|z_i-z_i^*\|]\le\sum_{t=0}^{T-1}l_{g,1}\sqrt{\mathbb{E}[\|z_t-z_t^*\|^2]}\le l_{g,1}\sqrt{T}\sqrt{\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]}. \tag{D.30}$$

For the second term on right-hand side of (D.29), we have

$$(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}L_{x,1}\|y_i-y_i^*\|\mathbb{E}\|\nabla\Phi(x_i)\|\le(1-\beta)\frac{L_{x,1}}{8L_1}\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\mathbb{E}\|\nabla\Phi(x_i)\|\le\frac{1}{8}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|, \tag{D.31}$$

where we use $\|y_t-y_t^*\|\le 1/(8L_1)$ for any $t\in[T]$ by Lemma D.7 and the fact that $L_{x,1}\le L_1$.
For the third term on right-hand side of (D.29), by (D.18) we have

$$\begin{aligned}&(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\|y_i-y_i^*\|\\&\le\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\left[\frac{4\Delta_{y,0}}{\mu\alpha-4(1-\beta)}+T\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right].\end{aligned} \tag{D.32}$$

Plugging (D.30), (D.31) and (D.32) into (D.29), and then combining (D.29) with (D.28) yields the upper bound:

$$\begin{aligned}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right]\le&\left(\frac{\eta L_1\beta}{1-\beta}+\frac{1}{8}\right)\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|+T\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2+\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2}+\frac{T\eta L_0\beta}{1-\beta}\\&+\sqrt{T}\left(\sqrt{2}\sigma_{g,2}\sqrt{1-\beta}+l_{g,1}\right)\sqrt{\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]}+\frac{\beta}{1-\beta}\|m_0-\nabla\Phi(x_0)\|\\&+\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\left[\frac{4\Delta_{y,0}}{\mu\alpha-4(1-\beta)}+T\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right].\end{aligned}$$

□

D.6. Proof of Theorem 4.1

Before statement of Theorem 4.1, we first modify Lemma C.4 to give a characterization for the objective function value decrease in terms of the true hypergradient $\nabla\Phi(x_t)$ and the moving average hypergradient estimation error ϵ_t. Similar results also appear in (Jin et al., 2021), (Liu et al., 2023) and (Hao et al., 2024).

Lemma D.12. *Consider an algorithm that starts at x_0 and updates via $x_{t+1}=x_t-\eta\frac{m_{t+1}}{\|m_{t+1}\|}$, where $\{m_t\}$ is any arbitrary sequence of points. Define $\epsilon_t:=m_{t+1}-\nabla\Phi(x_t)$ to be the estimation error. Then for any η satisfying*

$$\eta\le\frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}, \tag{D.33}$$

we have the following one-step improvement:

$$\Phi(x_{t+1})-\Phi(x_t)\le-\left(\eta-\frac{1}{2}L_1\eta^2\right)\|\nabla\Phi(x_t)\|+\frac{1}{2}L_0\eta^2+2\eta\|\epsilon_t\|. \tag{D.34}$$

Moreover, by a telescope sum and total expectation (taken over the randomness in $\widetilde{\mathcal{F}}_T$) we have

$$\left(1-\frac{1}{2}\eta L_1\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|\le\frac{\Phi(x_0)-\Phi(x_t)}{T\eta}+\frac{1}{2}\eta L_0+\frac{2}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right]. \tag{D.35}$$

Proof of Lemma D.12. Since η satisfies (D.33), we apply Lemma C.3 and C.4 with $x = x_{t+1}$ and $x' = x_t$ to obtain

$$\begin{aligned}
\Phi(x_{t+1}) &\le \Phi(x_t) + \langle \nabla\Phi(x_t), x_{t+1} - x_t\rangle + \frac{L_0 + L_1\|\nabla\Phi(x_t)\|}{2}\|x_{t+1} - x_t\|^2 \\
&= \Phi(x_t) - \eta\langle \nabla\Phi(x_t), \frac{m_{t+1}}{\|m_{t+1}\|}\rangle + \frac{1}{2}\eta^2(L_0 + L_1\|\nabla\Phi(x_t)\|) \\
&= \Phi(x_t) - \eta\langle m_{t+1} - \epsilon_t, \frac{m_{t+1}}{\|m_{t+1}\|}\rangle + \frac{1}{2}\eta^2(L_0 + L_1\|\nabla\Phi(x_t)\|) \\
&= \Phi(x_t) - \eta\|m_{t+1}\| + \eta\langle \epsilon_t, \frac{m_{t+1}}{\|m_{t+1}\|}\rangle + \frac{1}{2}\eta^2(L_0 + L_1\|\nabla\Phi(x_t)\|) \\
&\overset{(i)}{\le} \Phi(x_t) - \eta\|m_{t+1}\| + \eta\|\epsilon_t\| + \frac{1}{2}\eta^2(L_0 + L_1\|\nabla\Phi(x_t)\|) \\
&\overset{(ii)}{\le} \Phi(x_t) - \eta\|\nabla\Phi(x_t)\| + 2\eta\|\epsilon_t\| + \frac{1}{2}\eta^2(L_0 + L_1\|\nabla\Phi(x_t)\|)
\end{aligned} \tag{D.36}$$

where we use Cauchy-Schwarz inequality for (i) and $\|m_{t+1}\| = \|\nabla\Phi(x_t) + \epsilon_t\| \ge \|\nabla\Phi(x_t)\| - \|\epsilon_t\|$ for (ii). Rearranging it gives (D.34). Moreover, dividing $1/(T\eta)$ on both sides of (D.34), then taking telescope sum and total expectation yields (D.35). □

With Lemma D.11 and D.12, we are now ready to prove Theorem 4.1.

Theorem D.13 (Restatement of Theorem 4.1)**.** *Suppose Assumptions 3.1 and 3.2 hold. Let $\{x_t\}$ be the iterates produced by Algorithm 2. For any given $\delta \in (0,1)$ and any small ϵ satisfying*

$$\begin{aligned}
\epsilon \le \min\Bigg\{ & \frac{L_0}{L_1}, \Delta_{y,0}L_0, \frac{8l_{g,1}L_0}{\mu\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2 + L_{y,1}^2)}}, \sqrt{\frac{16el_{g,1}\Delta_0 L_0}{\mu\delta}}, 4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}; \\
& \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu/(2l_{g,1}))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu l_{g,1}\Delta_{z,0}/(2\Delta_0 L_0))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}}, \\
& \frac{\Delta_0 L_0}{\|\nabla\Phi(x_0)\|}, \frac{L_0\sigma_{g,1}}{\sigma_{g,2}}, \frac{L_0\sigma_{g,1}}{\sqrt{\mu l_{g,1}}}, \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}\exp\left(\frac{-l_{g,1}\sqrt{\sigma_{f,1}^2 + 2l_{f,0}^2\sigma_{g,2}^2/\mu^2}}{512L_0\sigma_{g,1}}\right)\Bigg\},
\end{aligned} \tag{D.37}$$

if we choose parameters $\alpha, \beta, \gamma, \eta$ as

$$\begin{aligned}
1-\beta = \min\Bigg\{ & 1, \frac{\mu}{16l_{g,1}}, \frac{16el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2}, \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}; \frac{\min\{1,\mu^2/(32l_{g,1}^2)\}}{\sigma_{f,1}^2 + 2l_{f,0}^2\sigma_{g,2}^2/\mu^2}\epsilon^2, \frac{l_{g,1}^2}{8\sigma_{g,2}^2}, \frac{\mu^2}{16\sigma_{g,2}^2}; \\
& \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu/(2l_{g,1}))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu l_{g,1}\Delta_{z,0}^2/(2\Delta_0 L_0))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}\Bigg\}
\end{aligned} \tag{D.38}$$

$$\eta = \min\left\{\frac{1}{8}\min\left(\frac{1}{L_1}, \frac{\epsilon}{L_0}, \frac{\epsilon\Delta_0}{\Delta_{y,0}^2 L_0^2}, \frac{\Delta_0}{\|\nabla\Phi(x_0)\|}, \frac{\epsilon\Delta_0}{l_{g,1}^2\Delta_{z,0}^2}, \frac{\mu\epsilon}{l_{g,1}L_0\log(A)}\right)(1-\beta), \frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2 + L_{y,1}^2)}}\right\}, \tag{D.39}$$

$$\gamma = \frac{1}{\mu}(1-\beta), \quad \alpha = 8\gamma, \tag{D.40}$$

where $\Delta_0, \Delta_{y,0}, \Delta_{z,0}, A, B$ are defined in (D.13), then with probability at least $1-2\delta$ over the randomness in $\sigma(\widetilde{\mathcal{F}}_{T_0}^1 \cup \mathcal{F}_T^1)$, Algorithm 2 guarantees $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \le 14\epsilon$ with at most $T = \frac{4\Delta_0}{\eta\epsilon}$ iterations, where the expectation is taken over over the randomness in $\widetilde{\mathcal{F}}_T$.

Proof of Theorem D.13. Before we begin the proof, let's first briefly describe the simple motivation behind the seemingly complex choice of parameters: we aim to choose $\epsilon, \alpha, \beta, \gamma, \eta$ carefully such that all of the conditions needed for Lemma C.3, C.4, D.8 and D.7 hold. Especially for Lemma D.7, we need to choose suitable β and ϵ such that the following two terms in (D.39) and (D.38) dominant (please check Lemma D.14 for more details):

$$\eta = \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta) \qquad \text{and} \qquad 1-\beta = \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}, \tag{D.41}$$

and this is primarily where those "lengthy" formulas come from. In addition, we also need to choose proper β, η and total number of iterations T such that (D.44), (D.45) and (D.46) (those three terms are mainly form Lemma D.11) can be controlled and hence small enough to guarantee the convergence of Algorithm 2. With this in hand, now we start proof.

By Lemma D.12 and definition (D.13) of Δ_0, we have the following estimate:

$$\left(1-\frac{1}{2}\eta L_1\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \le \frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\epsilon_t\|\right]. \tag{D.42}$$

Under event $\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y$, plug (D.23) into the above inequality, then we have

$$\left(1-\left(\frac{1}{2}+\frac{2\beta}{1-\beta}\right)\eta L_1-\frac{1}{4}\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \tag{D.43}$$

$$\le 2\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2+\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2}+\frac{2\eta L_0\beta}{1-\beta}+\frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2\beta}{T(1-\beta)}\|m_0-\nabla\Phi(x_0)\| \tag{D.44}$$

$$+2\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\left[\frac{4\Delta_{y,0}}{T(\mu\alpha-4(1-\beta))}+\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right] \tag{D.45}$$

$$+2\left(\sqrt{2}\sigma_{g,2}\sqrt{1-\beta}+l_{g,1}\right)\sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]}. \tag{D.46}$$

Now we proceed to bound (D.43), (D.44), (D.45) and (D.46), respectively.

For left-hand side (D.43) of the inequality, we have

$$\left(1-\left(\frac{1}{2}+\frac{2\beta}{1-\beta}\right)\eta L_1-\frac{1}{4}\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \ge \frac{1}{2T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|, \tag{D.47}$$

since the following holds

$$1-\left(\frac{1}{2}+\frac{2\beta}{1-\beta}\right)\eta L_1-\frac{1}{4}=1-\frac{1+3\beta}{2(1-\beta)}\eta L_1-\frac{1}{4}\ge 1-\frac{2\eta L_1}{1-\beta}-\frac{1}{4}\ge\frac{1}{2},$$

where we use $\beta\le 1$ for the first inequality and $\eta\le(1-\beta)/(8L_1)$ for the second inequality.

For the first term (D.44) on right-hand side of the inequality, we have

$$\begin{aligned}&2\sqrt{1-\beta}\sqrt{\sigma_{f,1}^2+2\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2}+\frac{2\eta L_0\beta}{1-\beta}+\frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2\beta}{T(1-\beta)}\|m_0-\nabla\Phi(x_0)\|\\ &\overset{(i)}{\le} 2\epsilon+\frac{1}{4}\epsilon+\frac{1}{4}\epsilon+\frac{1}{16}\epsilon+\frac{\eta\epsilon\beta}{2(1-\beta)\Delta_0}\|\nabla\Phi(x_0)\|\overset{(ii)}{\le} 2\epsilon+\frac{1}{4}\epsilon+\frac{1}{4}\epsilon+\frac{1}{16}\epsilon+\frac{1}{16}\epsilon=\frac{21}{8}\epsilon.\end{aligned} \tag{D.48}$$

where (i) and (ii) follow from the choice of β and η as in (D.38) and (D.39) that

$$1-\beta\le\frac{\epsilon^2}{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2},\quad \eta\le\frac{\epsilon}{8L_0}(1-\beta),\quad T=\frac{4\Delta_0}{\eta\epsilon},\quad m_0=0;\quad \eta\le\frac{\Delta_0}{8\|\nabla\Phi(x_0)\|}(1-\beta).$$

For the second term (D.45) on right-hand side of the inequality, we have

$$\begin{aligned}&2\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\left[\frac{4\Delta_{y,0}}{T(\mu\alpha-4(1-\beta))}+\sqrt{\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2 l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)}\right]\\ &\overset{(i)}{\le} 2L_0\left(\frac{\Delta_{y,0}}{T(1-\beta)}+\sqrt{\frac{\epsilon^2}{512L_0^2}}\right)\overset{(ii)}{=} 2L_0\left(\frac{\eta\epsilon\Delta_{y,0}}{4\Delta_0(1-\beta)}+\frac{\epsilon}{16\sqrt{2}L_0}\right)\overset{(iii)}{\le} 2L_0\left(\frac{\epsilon}{32L_0}+\frac{\epsilon}{16\sqrt{2}L_0}\right)\le\frac{3}{16}\epsilon,\end{aligned} \tag{D.49}$$

where (i) follows from $\mu\alpha=8(1-\beta)$ and (D.14), (D.17) in Lemma D.7, (ii) follows from the choice of $T=4\Delta_0/(\eta\epsilon)$ and (iii) follows from $\eta\le\epsilon\Delta_0(1-\beta)/(8\Delta_{y,0}^2L_0^2)\le\Delta_0(1-\beta)/(8\Delta_{y,0}L_0)$ by $\epsilon\le\Delta_{y,0}L_0$ from (D.37).

For the third term (D.46) on right-hand side of the inequality, we have

$$
\begin{aligned}
&2\left(\sqrt{2}\sigma_{g,2}\sqrt{1-\beta}+l_{g,1}\right)\sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]} \le 3l_{g,1}\sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|z_t-z_t^*\|^2]} \\
&\le 3l_{g,1}\sqrt{\frac{1}{T\mu\gamma}\|z_0-z_0^*\|^2+\frac{10(1-\mu\gamma)}{T\mu^3(\alpha-2\gamma)}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\|y_0-y_0^*\|^2} \\
&\quad+3l_{g,1}\sqrt{\frac{2\gamma}{\mu}\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2+\sigma_{f,1}^2\right)+\frac{4l_{z^*}^2}{\mu^2}\frac{\eta^2}{\gamma^2}+\frac{5}{\mu^2}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\left(\frac{eT}{\delta}\right)} \\
&\overset{(i)}{\le} 3\sqrt{\frac{l_{g,1}^2}{T(1-\beta)}\|z_0-z_0^*\|^2+\frac{5(1-\mu\gamma)}{3T(1-\beta)}\left[\frac{l_{g,1}^2}{\mu^2}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\right]\|y_0-y_0^*\|^2} \\
&\quad+3\sqrt{\frac{2l_{g,1}^2}{\mu^2}\left(\frac{2l_{f,0}^2}{\mu^2}\sigma_{g,2}^2+\sigma_{f,1}^2\right)(1-\beta)+\frac{4l_{g,1}^2l_{z^*}^2}{\mu^2}\frac{\eta^2}{\gamma^2}+\left[\frac{5l_{g,1}^2}{\mu^2}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\right]\left(\frac{8\alpha\sigma_{g,1}^2}{\mu}+\frac{4\eta^2l_{g,1}^2}{\mu^4\alpha^2}\right)\log\frac{eT}{\delta}} \\
&\overset{(ii)}{\le} 3\sqrt{\frac{l_{g,1}^2\eta\epsilon\Delta_{z,0}^2}{4\Delta_0(1-\beta)}+\frac{5\eta\epsilon L_0^2\Delta_{y,0}^2}{12\Delta_0(1-\beta)}}+3\sqrt{\frac{1}{16}\epsilon^2+\frac{4L_0^2\eta^2}{(1-\beta)^2}+5L_0^2\frac{\epsilon^2}{128L_0^2}} \overset{(iii)}{\le} 3\sqrt{\frac{\epsilon^2}{32}+\frac{5\epsilon^2}{96}}+3\sqrt{\frac{\epsilon^2}{16}+\frac{\epsilon^2}{16}+\frac{5\epsilon^2}{128}} \le 4\epsilon,
\end{aligned}
\tag{D.50}
$$

where (i) follows from $\mu\gamma = 1-\beta$ and $\alpha = 8\gamma$, (ii) follows from (D.14), (D.17) in Lemma D.7 and the fact that

$$
T=\frac{4\Delta_0}{\eta\epsilon},\quad \frac{l_{g,1}^2}{\mu^2}\left(\frac{l_{g,2}^2l_{f,0}^2}{\mu^2}+(L_{y,0}+L_{y,1}l_{f,0})^2\right)\le L_0^2,\quad 1-\beta\le\frac{\mu^2\epsilon^2}{32l_{g,1}^2(\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2)},\quad l_{g,1}^2l_{z^*}^2\le L_0^2,\quad \mu\gamma=1-\beta,
$$

and (iii) follows from the choice of η as in (D.39):

$$
\eta\le\frac{\epsilon\Delta_0}{8l_{g,1}^2\Delta_{z,0}^2}(1-\beta),\quad \eta\le\frac{\epsilon\Delta_0}{8\Delta_{y,0}^2L_0^2}(1-\beta),\quad \eta\le\frac{\epsilon}{8L_0}(1-\beta).
$$

Combining (D.47), (D.48), (D.49) and (D.50) together yields

$$
\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|\le 2\left(\frac{21}{8}\epsilon+\frac{3}{16}\epsilon+4\epsilon\right)\le 14\epsilon.
$$

Also note that

$$
\Pr(\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y)=\Pr(\mathcal{E}_y\mid\mathcal{E}_{\mathsf{init}})\cdot\Pr(\mathcal{E}_{\mathsf{init}})\ge(1-\delta)^2\ge 1-2\delta.
$$

Therefore, with probability at least $1-2\delta$ over the randomness in $\mathcal{F}_T^1$, we have $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|\le 14\epsilon$, where the expectation is taken over over the randomness in $\widetilde{\mathcal{F}}_T$. □

D.7. Omitted Proofs in Lemma D.7 and Theorem D.13

Lemma D.14. *Under the same parameter choice in Theorem D.13, we have the following facts:*

$$
\eta=\frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta)\qquad\text{and}\qquad 1-\beta=\frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)},
$$

where A and B are defined in (D.13).

Proof of Lemma D.14. Let us verify this fact respectively.

Verification for η. First, we have

$$
\epsilon\le\min\left\{\frac{L_0}{L_1},\frac{\Delta_0L_0}{\|\nabla\Phi(x_0)\|},\frac{8l_{g,1}L_0}{\mu\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}\right\}
$$

which implies

$$\frac{1}{8L_1}(1-\beta) \le \frac{\epsilon}{8L_0}(1-\beta), \quad \frac{\Delta_0}{8\|\nabla\Phi(x_0)\|} \le \frac{\epsilon}{8L_0}(1-\beta), \quad \frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}} \le \frac{\epsilon}{8L_0}(1-\beta).$$

Also we have

$$1-\beta \le \min\left\{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu/(2l_{g,1}))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu l_{g,1}\Delta_{z,0}^2/(2\Delta_0 L_0))}, \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}\right\},$$

which implies that

$$\frac{\epsilon}{8L_0}(1-\beta) \le \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta), \quad \frac{\epsilon\Delta_0}{8\Delta_{y,0}^2L_0^2}(1-\beta) \le \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta), \quad \frac{\epsilon\Delta_0}{l_{g,1}^2\Delta_{z,0}^2}(1-\beta) \le \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta).$$

Therefore, we conclude that

$$\eta = \frac{\mu\epsilon}{8l_{g,1}L_0\log(A)}(1-\beta).$$

Verification for $1-\beta$. First, we have

$$\epsilon \le \min\left\{\sqrt{\frac{16el_{g,1}\Delta_0 L_0}{\mu\delta}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu/(2l_{g,1}))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu l_{g,1}\Delta_{z,0}/(2\Delta_0 L_0))}}, \sqrt{\frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))}}\right\},$$

which implies that

$$\frac{16el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2} \ge 1, \quad \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu/(2l_{g,1}))} \ge 1, \quad \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu l_{g,1}\Delta_{z,0}^2/(2\Delta_0 L_0))} \ge 1, \quad \frac{32el_{g,1}\Delta_0 L_0}{\mu\delta\epsilon^2\exp(\mu\Delta_{y,0}^2 L_0/(2l_{g,1}\Delta_0))} \ge 1.$$

Also, we have

$$\epsilon \le \min\left\{4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}, \frac{L_0\sigma_{g,1}}{\sigma_{g,2}}, \frac{L_0\sigma_{g,1}}{\sqrt{\mu l_{g,1}}}\right\},$$

which implies that

$$B = \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta\epsilon^4}\right)^4 \ge 4 \implies \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)} \le \frac{\mu^2}{16\sigma_{g,2}^2} \le \frac{l_{g,1}^2}{8\sigma_{g,2}^2}, \quad \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)} \le \frac{\mu}{16l_{g,1}} < 1.$$

Finally,

$$\epsilon \le \left(\frac{2^{21}el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}\exp\left(\frac{-l_{g,1}\sqrt{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2}}{512L_0\sigma_{g,1}}\right)$$

implies that

$$\frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)} \le \frac{\mu^2/(32l_{g,1}^2)}{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2}\epsilon^2 = \frac{\min\{1,\mu^2/(32l_{g,1}^2)\}}{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2}\epsilon^2.$$

Therefore, we conclude that

$$1-\beta = \frac{\mu^2\epsilon^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}.$$

□

E. Omitted Proofs in Section 4.2.2

Remark: In this section, for the high probability proof, by a slight abuse of notation, we use $\mathbb{E}_t$ to denote the conditional expectation $\mathbb{E}[\cdot \mid \mathcal{F}_t^3]$.

E.1. Justification for Assumption 4.2

In this section, we provide justification for the last statement of Assumption 4.2. One example satisfying this assumption is that the random noise ζ is chosen based on the information of z. This makes sense in our setting because our algorithm access x, y, z first and then sample the random data to construct stochastic estimators. For example, we can choose ζ based on the following formula:

$$\|\nabla^2_{yy} G(x, y; \zeta) - \nabla^2_{yy} g(x, y)\| \overset{d}{=} \frac{\tau}{\|z\|}, \tag{E.1}$$

where $\overset{d}{=}$ means that the LHS and RHS have the same distribution, and τ can be any one dimensional bounded random variable, for instance, τ has a truncated normal (Gaussian) distribution lies within the interval $(-\sigma_z, \sigma_z)$, then we have

$$\forall \zeta, z,\ \|(\nabla^2_{yy} G(x, y; \zeta) - \nabla^2_{yy} g(x, y))z\| \le \sigma_z.$$

A specific example satisfying (E.1) is the following. Define the noise structure as $\nabla^2_{yy} G(x, y; \zeta) = \nabla^2_{yy} g(x, y) + \Gamma$, where $\Gamma = \mathrm{diag}(0, 0, \ldots, \tau/\|z\|)$.

E.2. Tracking the linear system solution: high-probability guarantees

In this section we follow the similar techniques as in Section D.1 to provide one-step improvement, distance recursion and distance tracking with high probability for estimator of the linear system solution, corresponding to Lemma E.1, E.2 and E.3, respectively. In particular, the proofs in this section are more involved than those in Section D.1 since we also need to handle error terms introduced by the lower-level variable besides variance and distribution drift. The key ideas are (i): to choose proper coefficient to apply Young's inequality and (ii): to make use of "good event", namely $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, to bound the additional error terms.

Now we start to prove one-step improvement and distance recursion for linear system estimator z_t.

Lemma E.1 (One-step improvement)**.** *Consider Algorithm 2 with sequence $\{z_t\}$ and constant learning rate $\gamma \le 1/(4l_{g,1})$, then for any z and $t \ge 1$, we have the following estimate:*

$$\begin{aligned} 2\gamma(h(x_t, z_{t+1}) - h(x_t, z)) \le{}& (1 - \mu\gamma)\|z_t - z\|^2 - \|z_{t+1} - z\|^2 + 2\gamma\langle v_t, z_t - z\rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 \\ &+ 8\gamma^2 l^2_{g,2}\|y_t - y^*_t\|^2\|z_t - z^*_t\|^2 + 8\gamma^2\frac{l^2_{g,2}l^2_{f,0}}{\mu^2}\|y_t - y^*_t\|^2 + 4\gamma^2(L_{y,0} + L_{y,1}l_{f,0})^2\|y_t - y^*_t\|^2, \end{aligned} \tag{E.2}$$

where we define v_t as

$$v_t = [\nabla^2_{yy} g(x_t, y_t) - \nabla^2_{yy} G(x_t, y_t; \zeta_t)]z_t - [\nabla_y f(x_t, y_t) - \nabla_y F(x_t, y_t; \xi_t)].$$

Proof of Lemma E.1. We define the objective function $h(x, z)$ as the following:

$$h(x, z) = \frac{1}{2}\langle \nabla^2_{yy} g(x, y^*(x))z, z\rangle - \langle \nabla_y f(x, y^*(x)), z\rangle.$$

Since h is $l_{g,1}$-smooth in z, we have

$$\begin{aligned} h(x_t, z_{t+1}) &\le h(x_t, z_t) + \langle \nabla^2_{yy} g(x_t, y^*_t)z_t - \nabla_y f(x_t, y^*_t), z_{t+1} - z_t\rangle + \frac{l_{g,1}}{2}\|z_{t+1} - z_t\|^2 \\ &\le h(x_t, z_t) + \langle \nabla^2_{yy} g(x_t, y_t)z_t - \nabla_y f(x_t, y_t), z_{t+1} - z_t\rangle + \frac{l_{g,1}}{2}\|z_{t+1} - z_t\|^2 \\ &\quad + \langle [\nabla^2_{yy} g(x_t, y^*_t) - \nabla^2_{yy} g(x_t, y_t)]z_t, z_{t+1} - z_t\rangle + \langle [\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y^*_t)], z_{t+1} - z_t\rangle \\ &\le h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t)z_t - \nabla_y F(x_t, y_t; \xi_t), z_{t+1} - z_t\rangle + \frac{l_{g,1}}{2}\|z_{t+1} - z_t\|^2 \\ &\quad + \langle [\nabla^2_{yy} g(x_t, y^*_t) - \nabla^2_{yy} g(x_t, y_t)]z_t, z_{t+1} - z_t\rangle + \langle [\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y^*_t)], z_{t+1} - z_t\rangle \\ &\quad + \langle [\nabla^2_{yy} g(x_t, y_t) - \nabla^2_{yy} G(x_t, y_t; \zeta_t)]z_t, z_{t+1} - z_t\rangle + \langle [\nabla_y F(x_t, y_t; \xi_t) - \nabla_y f(x_t, y_t)], z_{t+1} - z_t\rangle \\ &= h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t)z_t - \nabla_y F(x_t, y_t; \xi_t), z_{t+1} - z_t\rangle + \frac{l_{g,1}}{2}\|z_{t+1} - z_t\|^2 \\ &\quad + \langle [\nabla^2_{yy} g(x_t, y^*_t) - \nabla^2_{yy} g(x_t, y_t)]z_t, z_{t+1} - z_t\rangle + \langle [\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y^*_t)], z_{t+1} - z_t\rangle \\ &\quad + \langle v_t, z_{t+1} - z_t\rangle, \end{aligned} \tag{E.3}$$

where we define v_t as the following:

$$v_t = [\nabla^2_{yy} g(x_t, y_t) - \nabla^2_{yy} G(x_t, y_t; \zeta_t)] z_t - [\nabla_y f(x_t, y_t) - \nabla_y F(x_t, y_t; \xi_t)].$$

Next, given any $\delta_t > 0$, we apply Young's inequality to obtain

$$\langle v_t, z_{t+1} - z_t \rangle \leq \frac{\delta_t}{2} \|v_t\|^2 + \frac{1}{2\delta_t} \|z_{t+1} - z_t\|^2. \tag{E.4}$$

Also, we estimate the following by Young's inequality (with $\phi_t = 4\gamma$):

$$\begin{aligned}
\langle \nabla^2_{yy} g(x_t, y_t^*) z_t - \nabla^2_{yy} g(x_t, y_t) z_t, z_{t+1} - z_t \rangle &\leq \frac{\phi_t}{2} \|\nabla^2_{yy} g(x_t, y_t^*) - \nabla^2_{yy} g(x_t, y_t)\|^2 \|z_t\|^2 + \frac{1}{2\phi_t} \|z_{t+1} - z_t\|^2 \\
&\leq \frac{\phi_t}{2} l_{g,2}^2 \|y_t - y_t^*\|^2 (2\|z_t^*\|^2 + 2\|z_t - z_t^*\|^2) + \frac{1}{2\phi_t} \|z_{t+1} - z_t\|^2 \\
&\leq \frac{\phi_t}{2} l_{g,2}^2 \|y_t - y_t^*\|^2 \left(\frac{2 l_{f,0}^2}{\mu^2} + 2\|z_t - z_t^*\|^2 \right) + \frac{1}{2\phi_t} \|z_{t+1} - z_t\|^2 \\
&\leq 4\gamma l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 4\gamma \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} \|y_t - y_t^*\|^2 + \frac{(4\gamma)^{-1}}{2} \|z_{t+1} - z_t\|^2,
\end{aligned} \tag{E.5}$$

where we use $\|z_t^*\| = \|[\nabla^2_{yy} g(x, y^*(x))]^{-1} \nabla_y f(x, y^*(x))\| \leq l_{f,0}/\mu$ for the second inequality.

Again we apply Young's inequality to estimate (with $\varphi_t = 4\gamma$):

$$\begin{aligned}
\langle \nabla_y f(x_t, y_t) - \nabla_y f(x_t, y_t^*), z_{t+1} - z_t \rangle &\leq \frac{\varphi_t}{2} \|\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y_t^*)\|^2 + \frac{1}{2\varphi_t} \|z_{t+1} - z_t\|^2 \\
&\leq \frac{\varphi_t}{2} (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2 + \frac{1}{2\varphi_t} \|z_{t+1} - z_t\|^2 \\
&\leq 2\gamma (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2 + \frac{(4\gamma)^{-1}}{2} \|z_{t+1} - z_t\|^2.
\end{aligned} \tag{E.6}$$

Therefore, given any z, combining (E.3), (E.4), (E.5) and (E.6) we have

$$\begin{aligned}
h(x_t, z_{t+1}) &\leq h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z_{t+1} - z_t \rangle + \frac{l_{g,1}}{2} \|z_{t+1} - z_t\|^2 \\
&\quad + \langle [\nabla^2_{yy} g(x_t, y_t^*) - \nabla^2_{yy} g(x_t, y_t)] z_t, z_{t+1} - z_t \rangle + \langle [\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y_t^*)], z_{t+1} - z_t \rangle \\
&\quad + \langle v_t, z_{t+1} - z_t \rangle \\
&\leq h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z_{t+1} - z_t \rangle + \frac{\delta_t^{-1} + l_{g,1} + 2(4\gamma)^{-1}}{2} \|z_{t+1} - z_t\|^2 + \frac{\delta_t}{2} \|v_t\|^2 \\
&\quad + 4\gamma l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 4\gamma \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} \|y_t - y_t^*\|^2 + 2\gamma (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2 \\
&\leq h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z_{t+1} - z_t \rangle + \frac{1}{2\gamma} \|z_{t+1} - z_t\|^2 \\
&\quad + \frac{\delta_t^{-1} + l_{g,1} + (2\gamma)^{-1} - \gamma^{-1}}{2} \|z_{t+1} - z_t\|^2 + \frac{\delta_t}{2} \|v_t\|^2 \\
&\quad + 4\gamma l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 4\gamma \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} \|y_t - y_t^*\|^2 + 2\gamma (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2 \\
&\leq h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z - z_t \rangle + \frac{1}{2\gamma} \|z - z_t\|^2 - \frac{1}{2\gamma} \|z - z_{t+1}\|^2 \\
&\quad + \frac{\delta_t^{-1} + l_{g,1} - (2\gamma)^{-1}}{2} \|z_{t+1} - z_t\|^2 + \frac{\delta_t}{2} \|v_t\|^2 \\
&\quad + 4\gamma l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 4\gamma \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2} \|y_t - y_t^*\|^2 + 2\gamma (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2
\end{aligned}$$

where the last inequality holds since $z_{t+1} = z_t - \gamma[\nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t)]$ is the unique minimizer of the γ^{-1}-strongly convex function $l(z) = \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z - z_t \rangle + \frac{1}{2\gamma} \|z - z_t\|^2$, and thus

$l(z) - l(z_{t+1}) \geq \frac{1}{2\gamma}\|z - z_{t+1}\|^2$ holds for any z. Now by μ-strong convexity of $h(x, z)$ in terms of z, we estimate

$$\begin{aligned}
&h(x_t, z_t) + \langle \nabla^2_{yy} G(x_t, y_t; \zeta_t) z_t - \nabla_y F(x_t, y_t; \xi_t), z - z_t \rangle \\
&\quad = h(x_t, z) + \langle \nabla^2_{yy} g(x_t, y_t) z_t - \nabla_y f(x_t, y_t), z - z_t \rangle + \langle v_t, z_t - z \rangle \\
&\quad \leq h(x_t, z) - \frac{\mu}{2}\|z - z_t\|^2 + \langle v_t, z_t - z \rangle
\end{aligned}$$

Thus we have

$$\begin{aligned}
h(x_t, z_{t+1}) \leq\ & h(x_t, z) - \frac{\mu}{2}\|z - z_t\|^2 + \langle v_t, z_t - z \rangle + \frac{1}{2\gamma}\|z - z_t\|^2 - \frac{1}{2\gamma}\|z - z_{t+1}\|^2 \\
&+ \frac{\delta_t^{-1} + l_{g,1} - (2\gamma)^{-1}}{2}\|z_{t+1} - z_t\|^2 + \frac{\delta_t}{2}\|v_t\|^2 \\
&+ 4\gamma l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 4\gamma \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 2\gamma (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2
\end{aligned}$$

Finally, taking $\delta_t = 2\gamma/(1 - 2l_{g,1}\gamma)$ and rearranging yields

$$\begin{aligned}
2\gamma(h(x_t, z_{t+1}) - h(x_t, z)) \leq\ & (1 - \mu\gamma)\|z_t - z\|^2 - \|z_{t+1} - z\|^2 + 2\gamma\langle v_t, z_t - z \rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 \\
&+ 8\gamma^2 l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 8\gamma^2 \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 4\gamma^2 (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2,
\end{aligned}$$

which is as claimed. □

Lemma E.2 (Distance recursion)**.** *Consider Algorithm 2 with sequence $\{z_t\}$ and constant learning rate $\gamma \leq 1/(4l_{g,1})$, then under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, for any $t \geq 1$, we have the following recursion:*

$$\begin{aligned}
\|z_{t+1} - z_{t+1}^*\|^2 \leq\ & \left(1 - \frac{\mu\gamma}{2}\right)\|z_t - z_t^*\|^2 + 2\gamma\langle v_t, z_t - z_t^* \rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 + \left(1 + \frac{1}{\mu\gamma}\right)\|z_t^* - z_{t+1}^*\|^2 \\
&+ 8\gamma^2 \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 4\gamma^2 (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2.
\end{aligned} \tag{E.7}$$

Proof of Lemma E.2. Note that the μ-strong convexity implies

$$\frac{\mu}{2}\|z_{t+1} - z_t^*\|^2 \leq h(x_t, z_{t+1}) - h(x_t, z_t^*).$$

Combining this estimate with Lemma E.1 under the identification $z = z_t^*$ yields

$$\begin{aligned}
(1 + \mu\gamma)\|z_{t+1} - z_t^*\|^2 \leq\ & (1 - \mu\gamma)\|z_t - z_t^*\|^2 + 2\gamma\langle v_t, z_t - z_t^* \rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 \\
&+ 8\gamma^2 l_{g,2}^2 \|y_t - y_t^*\|^2 \|z_t - z_t^*\|^2 + 8\gamma^2 \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 4\gamma^2 (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2 \\
\leq\ & (1 - \mu\gamma + 8\gamma^2 l_{g,2}^2 \|y_t - y_t^*\|^2)\|z_t - z_t^*\|^2 + 2\gamma\langle v_t, z_t - z_t^* \rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 \\
&+ 8\gamma^2 \frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 4\gamma^2 (L_{y,0} + L_{y,1} l_{f,0})^2 \|y_t - y_t^*\|^2,
\end{aligned}$$

under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, for any $t \in [T]$ we have

$$8\gamma^2 l_{g,2}^2 \|y_t - y_t^*\|^2 \leq \frac{\gamma^2 l_{g,2}^2}{8L_1^2} \leq \frac{\mu\gamma}{2},$$

where for the first inequality we use Lemma D.7, and for the second inequality we use (E.12) and (E.10) that

$$\gamma = \frac{16}{\mu}(1 - \beta) \qquad \text{and} \qquad 1 - \beta \leq \frac{\mu^2 L_1^2}{4l_{g,2}^2}.$$

Then we have

$$\begin{aligned}(1+\mu\gamma)\|z_{t+1}-z_t^*\|^2 \le \left(1-\frac{\mu\gamma}{2}\right)\|z_t-z_t^*\|^2 + 2\gamma\langle v_t, z_t-z_t^*\rangle + \frac{2\gamma^2}{1-2l_{g,1}\gamma}\|v_t\|^2 \\ + 8\gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t-y_t^*\|^2 + 4\gamma^2(L_{y,0}+L_{y,1}l_{f,0})^2\|y_t-y_t^*\|^2,\end{aligned} \tag{E.8}$$

Next, under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$ and an application of Young's inequality combining with (E.8) reveals

$$\begin{aligned}\|z_{t+1}-z_{t+1}^*\|^2 &\le (1+\mu\gamma)\|z_{t+1}-z_t^*\|^2 + (1+(\mu\gamma)^{-1})\|z_t^*-z_{t+1}^*\|^2 \\ &\le \left(1-\frac{\mu\gamma}{2}\right)\|z_t-z_t^*\|^2 + 2\gamma\langle v_t, z_t-z_t^*\rangle + \frac{2\gamma^2}{1-2l_{g,1}\gamma}\|v_t\|^2 + \left(1+\frac{1}{\mu\gamma}\right)\|z_t^*-z_{t+1}^*\|^2 \\ &\quad + 8\gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t-y_t^*\|^2 + 4\gamma^2(L_{y,0}+L_{y,1}l_{f,0})^2\|y_t-y_t^*\|^2.\end{aligned}$$

□

In the following lemmas and theorems, we will use the following parameter settings. In particular, we choose

$$\begin{aligned}\epsilon = \epsilon' G, \quad \epsilon' \le \min\Bigg\{ &\frac{L_0}{L_1}, \Delta_{y,0}L_0, \frac{64 l_{g,1}\Delta_{z,0}L_0}{\sqrt{4El_{g,1}^2+l_{z^*}^2}}, \frac{8l_{g,1}L_0}{\mu\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}, \sqrt{\frac{16el_{g,1}\Delta_0 L_0}{\mu\delta}}, 4\left(\frac{el_{g,1}\Delta_0 L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}; \\ &\frac{\sigma_{g,1}L_0L_1}{l_{g,2}}, \sqrt{\frac{32el_{g,1}\Delta_0L_0}{\mu\delta\exp(\mu/(2l_{g,1}))}}, \sqrt{\frac{32el_{g,1}\Delta_0L_0}{\mu\delta\exp(\mu l_{g,1}\Delta_{z,0}/(2\Delta_0L_0))}}, \sqrt{\frac{32el_{g,1}\Delta_0L_0}{\mu\delta\exp(\mu\Delta_{y,0}^2L_0/(2l_{g,1}\Delta_0))}}, \\ &\frac{\Delta_0}{\Delta_{z,0}}, \frac{\Delta_0 L_0}{\|\nabla\Phi(x_0)\|}, \frac{L_0\sigma_{g,1}}{\sigma_{g,2}}, \frac{L_0\sigma_{g,1}}{\sqrt{\mu l_{g,1}}}, \left(\frac{2^{21}el_{g,1}\Delta_0L_0^3\sigma_{g,1}^2}{\mu^3\delta}\right)^{1/4}\exp\left(\frac{-l_{g,1}\sqrt{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2}}{512L_0\sigma_{g,1}}\right)\Bigg\},\end{aligned} \tag{E.9}$$

$$\begin{aligned}1-\beta = \min\Bigg\{ &1, \frac{\mu}{16l_{g,1}}, \frac{16el_{g,1}\Delta_0L_0}{\mu\delta\epsilon'^2}, \frac{\mu^2\epsilon'^2}{64\cdot 1024L_0^2\sigma_{g,1}^2\log^2(B)}; \frac{\min\{1,\mu^2/(32l_{g,1}^2)\}}{\sigma_{f,1}^2+2l_{f,0}^2\sigma_{g,2}^2/\mu^2}\epsilon'^2, \frac{l_{g,1}^2}{8\sigma_{g,2}^2}, \frac{\mu^2}{16\sigma_{g,2}^2}; \\ &\frac{\mu^2L_1^2}{4l_{g,2}^2}, \frac{32el_{g,1}\Delta_0L_0}{\mu\delta\epsilon'^2\exp(\mu/(2l_{g,1}))}, \frac{32el_{g,1}\Delta_0L_0}{\mu\delta\epsilon'^2\exp(\mu l_{g,1}\Delta_{z,0}^2/(2\Delta_0L_0))}, \frac{32el_{g,1}\Delta_0L_0}{\mu\delta\epsilon'^2\exp(\mu\Delta_{y,0}^2L_0/(2l_{g,1}\Delta_0))}\Bigg\},\end{aligned} \tag{E.10}$$

$$\eta = \min\left\{\frac{1}{8}\min\left(\frac{1}{L_1}, \frac{\epsilon'}{L_0}, \frac{\Delta_0}{\Delta_{z,0}L_0}, \frac{\epsilon'\Delta_0}{\Delta_{y,0}^2L_0^2}, \frac{\Delta_0}{\|\nabla\Phi(x_0)\|}, \frac{\epsilon'\Delta_0}{l_{g,1}^2\Delta_{z,0}^2}, \frac{\mu\epsilon'}{l_{g,1}L_0\log(A)}\right)(1-\beta), \frac{1}{\sqrt{2(1+l_{g,1}^2/\mu^2)(L_{x,1}^2+L_{y,1}^2)}}\right\}, \tag{E.11}$$

$$\alpha^{\mathsf{init}} = \min\left\{\frac{1}{2l_{g,1}}, \frac{\mu}{2048L_1^2\sigma_{g,1}^2\log(e/\delta)}\right\}, \quad T_0 = \frac{\log\left(256L_1^2\|y_0^{\mathsf{init}}-y_0^*\|^2\right)}{\log\left(2/(2-\mu\alpha^{\mathsf{init}})\right)}, \quad \gamma = \frac{16}{\mu}(1-\beta), \quad \alpha = \frac{8}{\mu}(1-\beta), \quad T = \frac{4\Delta_0}{\eta\epsilon'}, \tag{E.12}$$

where $\Delta_0, \Delta_{y,0}, \Delta_{z,0}, A, B$ are defined in (D.13), and E and G are defined as

$$E \coloneqq \max\left\{\frac{4\bar{\sigma}^2}{\sigma_{g,1}^2}, \frac{l_{g,2}^2l_{f,0}^2}{8\mu^2\sigma_{g,1}^2L_1^2}, \frac{L_0^2}{16\sigma_{g,1}^2L_1^2}\right\}, \tag{E.13}$$

$$G \coloneqq \frac{\mu}{16\sigma_{g,1}L_0}\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)+\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}}{8L_0}+\frac{13}{8}. \tag{E.14}$$

With Lemma E.2 and Proposition D.3, as well as the fact that event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y \in \sigma(\widetilde{\mathcal{F}}_{T_0}^1 \cup \mathcal{F}_T^1)$ are independent of event in $\mathcal{F}_t^2$ for any $t \in [T]$, we are able to leverage the following distance tracking result with high probability.

Lemma E.3 (High-probability distance tracking)**.** *Suppose that Assumption 3.2 holds, let $\{z_t\}$ be the iterates produced by Algorithm 2 with constant learning rate $\gamma \le 1/(4l_{g,1})$. Then under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, for any fixed $t \in [T]$ and $\delta \in (0,1)$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\mathcal{F}_t^2$:*

$$\|z_t-z_t^*\|^2 \le \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0-z_0^*\|^2 + \left[\frac{16\gamma\bar{\sigma}^2}{\mu}+\frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2}+\frac{4\gamma}{\mu}\frac{l_{g,2}^2l_{f,0}^2}{8\mu^2L_1^2}+\frac{4\gamma}{16\mu L_1^2}(L_{y,0}+L_{y,1}l_{f,0})^2\right]\log\left(\frac{e}{\delta}\right), \tag{E.15}$$

where we denote $\bar{\sigma} = \sigma_z + \sigma_{f,1}$. As a consequence, under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, for any given $\delta \in (0,1)$ and all $t \in [T]$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\mathcal{F}_T^2$:

$$\|z_t - z_t^*\|^2 \leq \left(1 - \frac{\mu\gamma}{4}\right)^t \|z_0 - z_0^*\|^2 + \left[\frac{16\gamma\bar{\sigma}^2}{\mu} + \frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2} + \frac{4\gamma}{\mu}\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{4\gamma}{16\mu L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2\right]\log\left(\frac{eT}{\delta}\right). \tag{E.16}$$

Proof of Lemma E.3. For simplicity, we define $\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y} \coloneqq \mathbb{E}[\cdot \mid \mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y]$, where events $\mathcal{E}_{\mathsf{init}}$ and $\mathcal{E}_y$ are defined in Lemma D.6 and Lemma D.7. Under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, Lemma E.2 in the regime $\gamma \leq 1/(4l_{g,1})$ directly yields

$$\begin{aligned}
\|z_{t+1} - z_{t+1}^*\|^2 &\leq \left(1 - \frac{\mu\gamma}{2}\right)\|z_t - z_t^*\|^2 + 2\gamma\langle v_t, r_t\rangle\|z_t - z_t^*\| + 4\gamma^2\|v_t\|^2 + \frac{2}{\mu\gamma}\|z_t^* - z_{t+1}^*\|^2 \\
&\quad + 8\gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2}\|y_t - y_t^*\|^2 + 4\gamma^2(L_{y,0} + L_{y,1}l_{f,0})^2\|y_t - y_t^*\|^2, \\
&\leq \left(1 - \frac{\mu\gamma}{2}\right)\|z_t - z_t^*\|^2 + 2\gamma\langle v_t, z_t - z_t^*\rangle + \frac{2\gamma^2}{1 - 2l_{g,1}\gamma}\|v_t\|^2 + \left(1 + \frac{1}{\mu\gamma}\right)\|z_t^* - z_{t+1}^*\|^2 \\
&\quad + \gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{\gamma^2}{16L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2.
\end{aligned}$$

where for the first inequality we set $r_t \coloneqq \frac{z_t - z_t^*}{\|z_t - z_t^*\|}$ if z_t is distinct from z_t^* and set it to zero otherwise, and for the second inequality we use Lemma D.7, namely $\|y_t - y_t^*\| \leq 1/(8L_1)$. The right-hand side has the form of a contraction factor, gradient noise, estimation error from y_t and drift. Now the goal is to control the moment generating function $\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y}[\exp(\lambda\|z_t - z_t^*\|)]$ through this recursion. We apply Proposition D.3, with $\mathcal{H}_t = \mathcal{F}_t^2$, $V_t = \|z_t - z_t^*\|^2$, $U_t = 2\gamma\langle v_t, r_t\rangle$, $X_t = 4\gamma^2\|v_t\|^2 + 2\|z_t^* - z_{t+1}^*\|^2/(\mu\gamma)$, $\alpha_t = 1 - \mu\gamma/2$, $\kappa_t = C$, $\sigma_t = 2\gamma(\sigma_z + \sigma_{f,1})$ and $\nu_t = 4\gamma^2(\sigma_z + \sigma_{f,1})^2 + 2l_{z^*}^2\eta^2/(\mu\gamma)$, where we define constant C as

$$C \coloneqq \gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{\gamma^2}{16L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2,$$

then yielding the estimate

$$\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y}\left[\exp\left(\lambda\|z_{t+1} - z_{t+1}^*\|^2\right)\right] \leq \exp\left[\lambda\left(4\gamma^2\bar{\sigma}^2 + \frac{2l_{z^*}^2\eta^2}{\mu\gamma} + R\right)\right]\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y}\left[\exp\left(\lambda\left(1 - \frac{\mu\gamma}{4}\right)\|z_t - z_t^*\|^2\right)\right] \tag{E.17}$$

for all

$$0 \leq \lambda \leq \min\left\{\frac{\mu}{16\gamma\bar{\sigma}^2}, \frac{1}{8\gamma^2\bar{\sigma}^2 + 4l_{z^*}^2\eta^2/(\mu\gamma)}\right\}.$$

where we denote $\bar{\sigma} = \sigma_z + \sigma_{f,1}$ for simplicity. We deduce the following by iterating the recursion (E.17):

$$\begin{aligned}
\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y}\left[\exp\left(\lambda\|z_t - z_t^*\|^2\right)\right] &\leq \exp\left[\lambda\left(1 - \frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \lambda\left(4\gamma^2\bar{\sigma}^2 + \frac{2l_{z^*}^2\eta^2}{\mu\gamma} + C\right)\sum_{i=0}^{t-1}\left(1 - \frac{\mu\gamma}{4}\right)^i\right] \\
&\leq \exp\left\{\lambda\left[\left(1 - \frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \frac{4}{\mu\gamma}\left(4\gamma^2\bar{\sigma}^2 + \frac{2l_{z^*}^2\eta^2}{\mu\gamma} + C\right)\right]\right\} \\
&\leq \exp\left\{\lambda\left[\left(1 - \frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \frac{4}{\mu\gamma}\left(4\gamma^2\bar{\sigma}^2 + \frac{2l_{z^*}^2\eta^2}{\mu\gamma} + \gamma^2\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{\gamma^2}{16L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2\right)\right]\right\} \\
&\leq \exp\left\{\lambda\left[\left(1 - \frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \frac{16\gamma\bar{\sigma}^2}{\mu} + \frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2} + \frac{4\gamma}{\mu}\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{4\gamma}{16\mu L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2\right]\right\}
\end{aligned}$$

for all

$$0 \leq \lambda \leq \min\left\{\frac{\mu}{16\gamma\bar{\sigma}^2}, \frac{1}{8\gamma^2\bar{\sigma}^2 + 4l_{z^*}^2\eta^2/(\mu\gamma)}\right\}.$$

Moreover, setting

$$\nu \coloneqq \frac{16\gamma\bar{\sigma}^2}{\mu} + \frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2} + \frac{4\gamma}{\mu}\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2 L_1^2} + \frac{4\gamma}{16\mu L_1^2}(L_{y,0} + L_{y,1}l_{f,0})^2$$

and taking into account $\mu\gamma \leq 1$, we have

$$0 \leq \lambda \leq \frac{1}{\nu} \leq \min\left\{\frac{\mu}{16\gamma\bar{\sigma}^2}, \frac{1}{8\gamma^2\bar{\sigma}^2 + 4l_{z^*}^2\eta^2/(\mu\gamma)}\right\}.$$

Hence we obtain

$$\mathbb{E}_{|\mathcal{E}_{\mathsf{init}}\cap\mathcal{E}_y}\left[\exp\left[\lambda\left(\|z_t - z_t^*\|^2 - \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2\right)\right]\right] \leq \exp(\lambda\nu), \quad \forall 0 \leq \lambda \leq 1/\nu.$$

Taking $\lambda = 1/\nu$ and applying Markov's inequality yields that, for any given $\delta \in (0,1)$, under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, with probability at least $1-\delta$ over the randomness in $\mathcal{F}_t^2$:

$$\|z_t - z_t^*\|^2 \leq \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \left[\frac{16\gamma\bar{\sigma}^2}{\mu} + \frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2} + \frac{4\gamma}{\mu}\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2L_1^2} + \frac{4\gamma}{16\mu L_1^2}(L_{y,0}+L_{y,1}l_{f,0})^2\right]\log\left(\frac{e}{\delta}\right),$$

as claimed in (E.15). We obtain (E.16) by applying union bound. □

With Lemma E.1, E.2 and E.3, under suitable parameter choice as in (E.9), (E.10), (E.11) and (E.12), we are now able to leverage Lemma E.4 which provides refined control for z_t, and is also similar to what we did in Lemma D.7.

Lemma E.4. *Under Assumptions 3.1, 3.2 and the parameter setting (E.9), (E.10), (E.11) and (E.12), run Algorithm 2 for $T = \frac{4\Delta_0}{\eta\epsilon'}$ iterations. Then under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y$, for all $t \in [T]$ and any given $\delta \in (0,1)$, Algorithm 2 guarantees with probability at least $1-\delta$ over the randomness in $\mathcal{F}_T^2$ (we denote this event as $\mathcal{E}_z$) that:*

1. $\|z_t - z_t^*\| \leq 2\Delta_{z,0}$,
2. $\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|z_i - z_i^*\| \leq \left(1+\sqrt{E + l_{z^*}^2/(4l_{g,1}^2)}\right)\frac{\epsilon'}{32L_0}$,

where constant E is defined in (E.13).

Proof of Lemma E.4. We apply (E.16) and (E.12) to obtain

$$\begin{aligned}
\|z_t - z_t^*\|^2 &\leq \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \left[\frac{16\gamma\bar{\sigma}^2}{\mu} + \frac{8l_{z^*}^2\eta^2}{\mu^2\gamma^2} + \frac{4\gamma}{\mu}\frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2L_1^2} + \frac{4\gamma}{16\mu L_1^2}(L_{y,0}+L_{y,1}l_{f,0})^2\right]\log\left(\frac{eT}{\delta}\right)\\
&= \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \left[\frac{256(1-\beta)\bar{\sigma}^2}{\mu^2} + \frac{l_{z^*}^2\eta^2}{32\mu^2(1-\beta)^2} + \frac{8(1-\beta)}{\mu^2}\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2L_1^2} + \frac{4(1-\beta)}{\mu^2L_1^2}(L_{y,0}+L_{y,1}l_{f,0})^2\right]\log\left(\frac{eT}{\delta}\right)\\
&\leq \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \left(\frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2}E + \frac{\eta^2 l_{g,1}^2}{8\mu^2(1-\beta)^2}\frac{l_{z^*}^2}{4l_{g,1}^2}\right)\log\left(\frac{eT}{\delta}\right)\\
&\leq \left(1-\frac{\mu\gamma}{4}\right)^t\|z_0 - z_0^*\|^2 + \left(E + \frac{l_{z^*}^2}{4l_{g,1}^2}\right)\frac{\epsilon'^2}{1024L_0^2},
\end{aligned}$$

where for the first equality we use (E.12), for the second inequality we use conclusion of step 3 in Lemma D.7 to deduce

$$\max\left\{\frac{256(1-\beta)\bar{\sigma}^2}{\mu^2}, \frac{8(1-\beta)}{\mu^2}\frac{l_{g,2}^2 l_{f,0}^2}{\mu^2L_1^2}, \frac{4(1-\beta)}{\mu^2L_1^2}(L_{y,0}+L_{y,1}l_{f,0})^2\right\} \leq \frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2}E \leq \frac{\eta^2 l_{g,1}^2}{16\mu^2(1-\beta)^2}E,$$

with constant E defined as

$$E \coloneqq \max\left\{\frac{4\bar{\sigma}^2}{\sigma_{g,1}^2}, \frac{l_{g,2}^2 l_{f,0}^2}{8\mu^2\sigma_{g,1}^2L_1^2}, \frac{L_0^2}{16\sigma_{g,1}^2L_1^2}\right\},$$

and for the last inequality we apply (D.17) in Lemma D.7.

By (E.9), we have

$$\epsilon' \leq \frac{64 l_{g,1}\Delta_{z,0}L_0}{\sqrt{4El_{g,1}^2 + l_{z^*}^2}}$$

which implies for all $t \in [T]$ that

$$\|z_t - z_t^*\|^2 \le \left(1 - \frac{\mu\gamma}{4}\right)^t \|z_0 - z_0^*\|^2 + \left(E + \frac{l_{z^*}^2}{4l_{g,1}^2}\right) \frac{\epsilon'^2}{1024L_0^2} \le 2\Delta_{z,0}^2 \quad \Longrightarrow \quad \|z_t - z_t^*\| \le 2\Delta_{z,0},$$

thus the first part of the result is as claimed.

As for the second part, we have

$$\begin{aligned}
\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|z_i - z_i^*\| &\le \frac{(1-\beta)}{T}\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\sqrt{\left(1-\frac{\mu\gamma}{4}\right)^i\|z_0 - z_0^*\|^2 + \left(E + \frac{l_{z^*}^2}{4l_{g,1}^2}\right)\frac{\epsilon'^2}{1024L_0^2}} \\
&\le \frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\left[\left(1-\frac{\mu\gamma}{4}\right)^{i/2}\|z_0 - z_0^*\| + \sqrt{E + \frac{l_{z^*}^2}{4l_{g,1}^2}}\frac{\epsilon'}{32L_0}\right] \\
&\le \frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\left[\beta^t\sum_{i=0}^{t}\left(\frac{\sqrt{1-\mu\gamma/4}}{\beta}\right)^i\|z_0 - z_0^*\| + \frac{1}{1-\beta}\sqrt{E + \frac{l_{z^*}^2}{4l_{g,1}^2}}\frac{\epsilon'}{32L_0}\right] \\
&\le \frac{8\Delta_{z,0}}{T(\mu\gamma - 8(1-\beta))} + \sqrt{E + \frac{l_{z^*}^2}{4l_{g,1}^2}}\frac{\epsilon'}{32L_0},
\end{aligned} \tag{E.18}$$

where in the last inequality we use

$$\begin{aligned}
(1-\beta)\sum_{t=0}^{T-1}\beta^t\sum_{i=0}^{t}\left(\frac{\sqrt{1-\mu\gamma/4}}{\beta}\right)^i &\le (1-\beta)\sum_{t=0}^{T-1}\beta^t\frac{\beta}{\beta - \sqrt{1-\mu\gamma/4}} \le \frac{\beta}{\beta - \sqrt{1-\mu\gamma/4}} \le \frac{\beta\left(\beta + \sqrt{1-\mu\gamma/4}\right)}{\mu\gamma/4 - (1-\beta^2)} \\
&\le \frac{2}{\mu\gamma/4 - (1-\beta)(1+\beta)} \le \frac{2}{\mu\gamma/4 - 2(1-\beta)} = \frac{8}{\mu\gamma - 8(1-\beta)}.
\end{aligned}$$

Moreover, we have

$$\frac{8\Delta_{z,0}}{T(\mu\alpha - 8(1-\beta))} \le \frac{\Delta_{z,0}}{T(1-\beta)} = \frac{\eta\epsilon'\Delta_{z,0}}{4\Delta_0(1-\beta)} \le \frac{\epsilon'}{32L_0}, \tag{E.19}$$

where in the last inequality we use (E.11) that

$$\eta \le \frac{\Delta_0}{8\Delta_{z,0}L_0}(1-\beta).$$

Combining (E.18) and (E.19) yields

$$\frac{1}{T}(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\|z_i - z_i^*\| \le \frac{8\Delta_{z,0}}{T(\mu\gamma - 8(1-\beta))} + \sqrt{E + \frac{l_{z^*}^2}{4l_{g,1}^2}}\frac{\epsilon'}{32L_0} \le \left(1 + \sqrt{E + \frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{\epsilon'}{32L_0}.$$

□

Next, we proceed to give high probability bound for hypergradient estimation error, and we will use the following lemma in (Liu et al., 2023) as a technical tool.

Lemma E.5 (Lemma 2.4 in (Liu et al., 2023)). *Suppose $X_1, \ldots, X_T$ is a martingale difference sequence adapted to a filtration $F_1, F_2, \ldots$ in a Hilbert space such that $\|X_t\| \le R_t, \forall t \in [T]$ almost surely for some constant $R_t \ge 0$. Then for any given $\delta \in (0,1)$, with probability at least $1-\delta$, for all $t \in [T]$ we have*

$$\left\|\sum_{s=1}^{t} X_s\right\| \le 4\sqrt{\log(2/\delta)\sum_{s=1}^{T} R_s^2}.$$

With Lemma E.5, we are ready to give high probability bound for the following martingale difference sequence, which is one part of the hypergradient estimation error.

Lemma E.6. *Under Assumptions 3.1 and event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y \cap \mathcal{E}_z$, for any given $\delta \in (0,1)$ and fixed $t \in [T]$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\mathcal{F}_t^3$ (we denote this event as $\mathcal{E}_x$):*

$$\left\|\sum_{i=0}^{t} \beta^{t-i}\left(\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i') - \mathbb{E}_t[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')]\right)\right\| \le 4\left(\sigma_{f,1} + \frac{l_{f,0}\sigma_{g,2}}{\mu} + 2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\log(2/\delta)}{1-\beta}}. \tag{E.20}$$

As a consequence, under event $\mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y \cap \mathcal{E}_z$, for any given $\delta \in (0,1)$ and all $t \in [T]$, the following estimate holds with probability at least $1-\delta$ over the randomness in $\mathcal{F}_T^3$:

$$\left\|\sum_{i=0}^{t} \beta^{t-i}\left(\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i') - \mathbb{E}_t[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')]\right)\right\| \le 4\left(\sigma_{f,1} + \frac{l_{f,0}\sigma_{g,2}}{\mu} + 2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\log(2T/\delta)}{1-\beta}}. \tag{E.21}$$

Proof of Lemma E.6. By definition of $\widehat{\nabla}\Phi(x_t,y_t,z_t;\xi_t',\zeta_t')$, we have the following decomposition:

$$\begin{aligned}
&\widehat{\nabla}\Phi(x_t,y_t,z_t;\xi_t',\zeta_t') - \mathbb{E}_t[\widehat{\nabla}\Phi(x_t,y_t,z_t;\xi_t',\zeta_t')]\\
&= [\nabla_x F(x_t,y_t;\xi_t') - \nabla_{xy}^2 G(x_t,y_t,;\zeta_t')z_t] - [\nabla_x f(x_t,y_t) - \nabla_{xy}^2 g(x_t,y_t)]z_t\\
&= [\nabla_x F(x_t,y_t;\xi_t') - \nabla_x f(x_t,y_t)] - [\nabla_{xy}^2 G(x_t,y_t;\zeta_t') - \nabla_{xy}^2 g(x_t,y_t)]z_t\\
&= [\nabla_x F(x_t,y_t;\xi_t) - \nabla_x f(x_t,y_t)] + [\nabla_{xy}^2 G(x_t,y_t;\zeta_t) - \nabla_{xy}^2 g(x_t,y_t)]z_t^* - [\nabla_{xy}^2 G(x_t,y_t;\zeta_t) - \nabla_{xy}^2 g(x_t,y_t)](z_t - z_t^*).
\end{aligned}$$

For simplicity, we define $\mathcal{E}_{yz} := \mathcal{E}_{\mathsf{init}} \cap \mathcal{E}_y \cap \mathcal{E}_z$ and $\mathbb{E}_{|\mathcal{E}_{yz}}[\cdot] := \mathbb{E}[\cdot \mid \mathcal{E}_{yz}]$, where events $\mathcal{E}_{\mathsf{init}}$, $\mathcal{E}_y$ and $\mathcal{E}_z$ are defined in Lemma D.6, D.7 and E.4. Then for any $i \in [t]$, we have

$$\begin{aligned}
\beta^{t-i}[\nabla_x F(x_i,y_i;\xi_i') - \nabla_x f(x_i,y_i)] \in \mathcal{F}_{i+1}^3, \quad \mathbb{E}_{|\mathcal{E}_{yz}}\left[\beta^{t-i}[\nabla_x F(x_i,y_i;\xi_i') - \nabla_x f(x_i,y_i)] \mid \mathcal{F}_i^3\right] = 0;\\
\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)]z_i^* \in \mathcal{F}_{i+1}^3, \quad \mathbb{E}_{|\mathcal{E}_{yz}}\left[\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)]z_i^* \mid \mathcal{F}_i^3\right] = 0;\\
\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)](z_i - z_i^*) \in \mathcal{F}_{i+1}^3, \quad \mathbb{E}_{|\mathcal{E}_{yz}}\left[\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)](z_i - z_i^*) \mid \mathcal{F}_i^3\right] = 0.
\end{aligned}$$

Also by Assumption 4.2 and Lemma E.4, under event $\mathcal{E}_{yz} \in \sigma(\widetilde{\mathcal{F}}_{T_0}^1 \cup \mathcal{F}_T^1 \cup \mathcal{F}_T^2)$, the followings hold almost surely in $\mathcal{F}_t^3$:

$$\begin{aligned}
\|\beta^{t-i}[\nabla_x F(x_i,y_i;\xi_i') - \nabla_x f(x_i,y_i)]\| &\le \beta^{t-i}\sigma_{f,1}\\
\|\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)]z_i^*\| &\le \beta^{t-i}\sigma_{g,2}\frac{l_{f,0}}{\mu}\\
\|\beta^{t-i}[\nabla_{xy}^2 G(x_i,y_i;\zeta_i') - \nabla_{xy}^2 g(x_i,y_i)](z_i - z_i^*)\| &\le 2\beta^{t-i}\sigma_{g,2}\Delta_{z,0}
\end{aligned}$$

Thus under event $\mathcal{E}_{yz}$, for any $i \in [t]$, $\beta^{t-i}(\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i') - \mathbb{E}_i[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')])$ is a (almost surely) bounded martingale difference sequence. Now for any given $\delta \in (0,1)$, we apply Lemma E.5 to obtain under event $\mathcal{E}_{yz}$, with probability at least $1-\delta$ over the randomness in $\mathcal{F}_t^3$ that:

$$\begin{aligned}
\left\|\sum_{i=0}^{t} \beta^{t-i}\left(\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i') - \mathbb{E}_i[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')]\right)\right\| &\le 4\sqrt{\log(2/\delta)\sum_{i=0}^{t}\left[\beta^{t-i}\left(\sigma_{f,1} + \frac{l_{f,0}\sigma_{g,2}}{\mu} + 2\sigma_{g,2}\Delta_{z,0}\right)\right]^2}\\
&\le 4\left(\sigma_{f,1} + \frac{l_{f,0}\sigma_{g,2}}{\mu} + 2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\log(2/\delta)}{1-\beta}},
\end{aligned}$$

which is as claimed in (E.20). We obtain (E.21) by applying union bound. □

E.3. Proof of Theorem 4.3

By incorporating Lemma D.7, E.4 and E.6, we begin to prove Theorem 4.3.

Theorem E.7 (Restatement of Theorem 4.3)**.** *Suppose Assumptions 3.1 and 4.2 hold. Let $\{x_t\}$ be the iterates produced by Algorithm 2. For any given $\delta \in (0,1)$ and sufficiently small ϵ (see exact choice of ϵ in (E.9)), if we choose $\alpha^{\mathsf{init}}, \alpha, \beta, \gamma, \eta, T_0$ as (E.10), (E.11) and (E.12), then with probability at least $1-4\delta$ over the randomness in $\mathcal{F}_T$, Algorithm 2 guarantees $\frac{1}{T}\sum_{t=0}^{T}\|\nabla\Phi(x_t)\| \le \epsilon$ with at most $T = \frac{4\Delta_0}{\eta\epsilon}$ iterations.*

Proof of Theorem E.7. By similar approach as in Lemma D.11, we obtain

$$\begin{aligned}
\sum_{t=0}^{T-1}\|\epsilon_t\| &\le (1-\beta)\sum_{t=0}^{T-1}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')-\mathbb{E}_i[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')]\right)\right\|+\frac{\eta L_1\beta}{1-\beta}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|\\
&\quad+(1-\beta)\sum_{t=0}^{T-1}\left\|\sum_{i=0}^{t}\beta^{t-i}\left(\mathbb{E}_i[\widehat{\nabla}\Phi(x_i,y_i,z_i;\xi_i',\zeta_i')]-\nabla\Phi(x_i)\right)\right\|+\frac{T\eta L_0\beta}{1-\beta}+\frac{\beta}{1-\beta}\|m_0-\nabla\Phi(x_0)\|\\
&\le 4T(1-\beta)\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\log(2T/\delta)}{1-\beta}}+\frac{\eta L_1\beta}{1-\beta}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|\\
&\quad+(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}L_{x,1}\|y_i-y_i^*\|\|\nabla\Phi(x_i)\|+(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}\left(L_{x,0}+L_{x,1}\frac{l_{g,1}l_{f,0}}{\mu}+\frac{l_{g,2}l_{f,0}}{\mu}\right)\|y_i-y_i^*\| \qquad \text{(E.22)}\\
&\quad+(1-\beta)\sum_{t=0}^{T-1}\sum_{i=0}^{t}\beta^{t-i}l_{g,1}\|z_i-z_i^*\|+\frac{T\eta L_0\beta}{1-\beta}+\frac{\beta}{1-\beta}\|\nabla\Phi(x_0)\|\\
&\le 4T(1-\beta)\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\log(2T/\delta)}{1-\beta}}+\left(\frac{\eta L_1\beta}{1-\beta}+\frac{1}{8}\right)\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|+\frac{3T\epsilon'}{32}\\
&\quad+T\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}\epsilon'}{32L_0}+\frac{T\eta L_0\beta}{1-\beta}+\frac{\beta}{1-\beta}\|\nabla\Phi(x_0)\|
\end{aligned}$$

By (D.42) (without taking expectation), we have

$$\left(1-\frac{1}{2}\eta L_1\right)\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|\le\frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2}{T}\sum_{t=0}^{T-1}\|\epsilon_t\|.$$

Under event $\mathcal{E}_{\text{init}}\cap\mathcal{E}_y\cap\mathcal{E}_z$, plug (D.23) into the above inequality, then we have

$$\left(1-\left(\frac{1}{2}+\frac{2\beta}{1-\beta}\right)\eta L_1-\frac{1}{4}\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \qquad \text{(E.23)}$$

$$\le 8\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{(1-\beta)\log\left(\frac{2T}{\delta}\right)}+\frac{3\epsilon'}{16}+\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}\epsilon'}{16L_0} \qquad \text{(E.24)}$$

$$+\frac{2\eta L_0\beta}{1-\beta}+\frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2\beta}{T(1-\beta)}\|\nabla\Phi(x_0)\|. \qquad \text{(E.25)}$$

Now we proceed to bound (E.23), (E.24) and (E.25), respectively.

For left-hand side (E.23) of the inequality, by (D.47) we have

$$\left(1-\left(\frac{1}{2}+\frac{2\beta}{1-\beta}\right)\eta L_1-\frac{1}{4}\right)\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|\ge\frac{1}{2T}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|. \qquad \text{(E.26)}$$

For (E.24) on right-hand side of the inequality, we have

$$\begin{aligned}
8\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{(1-\beta)\log\left(\frac{2T}{\delta}\right)}&\le 8\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\sqrt{\frac{\mu^2}{64\sigma_{g,1}^2}\frac{\epsilon'^2}{1024L_0^2}}\\
&=\frac{\mu}{\sigma_{g,1}}\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\frac{\epsilon'}{32L_0},
\end{aligned} \qquad \text{(E.27)}$$

where for the first inequality we use (D.17) in Lemma D.7,

$$\frac{64(1-\beta)\sigma_{g,1}^2}{\mu^2}\log\left(\frac{eT}{\delta}\right)\le\frac{\epsilon'^2}{1024L_0^2}.$$

For (E.25) on right-hand side of the inequality, by (D.48) we have

$$\frac{2\eta L_0\beta}{1-\beta}+\frac{\Delta_0}{T\eta}+\frac{1}{2}\eta L_0+\frac{2\beta}{T(1-\beta)}\|\nabla\Phi(x_0)\|\le\frac{1}{4}\epsilon'+\frac{1}{4}\epsilon'+\frac{1}{16}\epsilon'+\frac{1}{16}\epsilon'=\frac{5}{8}\epsilon'. \qquad \text{(E.28)}$$

Combining (E.26), (E.27) and (E.28) together yields

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\| \le 2\left[\frac{\mu}{\sigma_{g,1}}\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)\frac{\epsilon'}{32L_0}+\frac{3\epsilon'}{16}+\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}\epsilon'}{16L_0}+\frac{5\epsilon'}{8}\right]$$
$$=\epsilon'\left[\frac{\mu}{16\sigma_{g,1}L_0}\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)+\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}}{8L_0}+\frac{13}{8}\right].$$

Recall constant G in (E.9) is defined as

$$G:=\frac{\mu}{16\sigma_{g,1}L_0}\left(\sigma_{f,1}+\frac{l_{f,0}\sigma_{g,2}}{\mu}+2\sigma_{g,2}\Delta_{z,0}\right)+\left(1+\sqrt{E+\frac{l_{z^*}^2}{4l_{g,1}^2}}\right)\frac{l_{g,1}}{8L_0}+\frac{13}{8}.$$

Again by (E.9), we have $\epsilon'=\epsilon/G$, thus we obtain $\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\Phi(x_t)\|\le\epsilon$. Also note that

$$\Pr(\mathcal{E}_x\cap\mathcal{E}_y\cap\mathcal{E}_z\cap\mathcal{E}_{\mathsf{init}})=\Pr(\mathcal{E}_x\mid\mathcal{E}_y\cap\mathcal{E}_z\cap\mathcal{E}_{\mathsf{init}})\cdot\Pr(\mathcal{E}_z\mid\mathcal{E}_y\cap\mathcal{E}_{\mathsf{init}})\cdot\Pr(\mathcal{E}_y\mid\mathcal{E}_{\mathsf{init}})\cdot\Pr(\mathcal{E}_{\mathsf{init}})\ge(1-\delta)^4\ge1-4\delta.$$

Therefore, with probability at least $1-4\delta$ over the randomness in $\mathcal{F}_T$, we have $\frac{1}{T}\sum_{t=0}^{T}\|\nabla\Phi(x_t)\|\le\epsilon$. □

F. Experimental Parameter Selection

F.1. Hyper-representation

In the experiments, we utilized grid search to find the best lower-level and upper-level learning rates within the scope of $(0.001, 0.5)$ for various methods. Specifically, on the SNLI dataset, optimal learning rate pairs are determined as $(0.01, 0.01)$ for MAML, $(0.01, 0.05)$ for ANIL, $(0.01, 0.01)$ for StocBio, $(0.02, 0.1)$ for TTSA, $(0.01, 0.05)$ for SABA, $(0.05, 0.05)$ for MA-SOBA, and $(0.05, 0.1)$ for both BO-REP and SLIP. On ARD dataset, the best combinations are $(0.05, 0.1)$ for both MAML and ANIL, $(0.05, 0.05)$ for StocBio, $(0.1, 0.01)$ for TTSA, $(0.05, 0.05)$ for SABA, $(0.05, 0.1)$ for MA-SOBA, $(0.001, 0.01)$ for BO-REP, and $(0.01, 0.1)$ for SLIP.

The following settings are applied to both SNLI and ARD datasets: For double-loop frameworks such as MAML, ANIL, and StocBio, the inner-loop iteration count is searched among 5, 10, and 20, where 5 is the best choice for these methods. For approaches SABA, MA-SOBA, BO-REP, and SLIP, the step size for the linear system variable z is consistently chosen as 0.01, based on the optimal parameter tuning from range of $(0.001, 0.1)$. For the fully first-order method F^{2}SA, which incorporates three distinct decision variables, corresponding learning rates are fine-tuned from a range of $(0.001, 0.5)$, with the best configuration being $(0.05, 0.05, 0.01)$. The momentum parameter β for MA-SOBA, BO-REP, and SLIP is set to 0.9. The Lagrangian multiplier λ in F^{2}SA is increased by 0.01 with each outer update. BO-REP updates the lower-level variable at every 2 outer intervals and conducts 3 iterations for each update. For SLIP, the number of warm-start steps for lower-level updates is set to 3. The batch size is set to 50 for all the methods.

F.2. Data Hyper-cleaning

In data hyper-cleaning, we also employ a grid search technique for all baseliens to determine the optimal lower-level and upper-level learning rates from the range of $(0.01, 0.1)$. The best learning rate pairings for lower-level and upper-level are $(0.05, 0.05)$ for StocBio, $(0.05, 0.01)$ for TTSA, $(0.1, 0.05)$ for both SABA and MA-SOBA, and $(0.05, 0.05)$ for BO-REP and SLIP. For the F^{2}SA algorithm, which manages three decision variables, the chosen learning rates were $(0.05, 0.05, 0.01)$. Additionally, the step size for addressing the linear system variable z in SABA, MA-SOBA, BO-REP, and SLIP is set as 0.01.

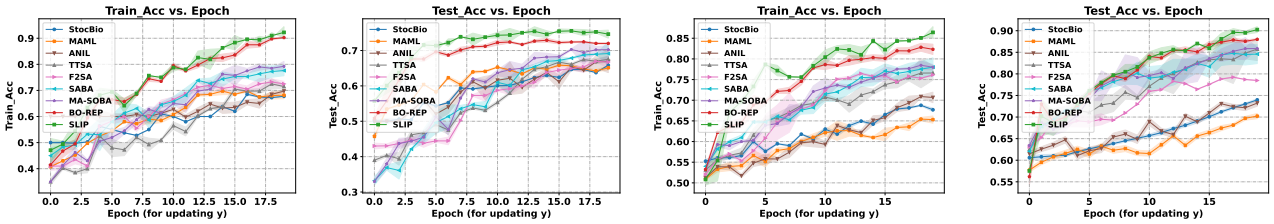
The number of inner loops in StocBio was fixed as 3, based on a selection range of $3, 5, 10$. For BO-REP, the update interval and the number of iterations per lower-level update are consistently set at 2 and 3, respectively. A uniform batch size of 512 was applied for all baselines. Other key experimental settings, such as the momentum parameters for MA-SOBA, BO-REP, and SLIP,as well as the increment of the Lagrangian multiplier in F^{2}SA, are the same as the specifications detailed in Section F.1.

G. Experiments with a Revised Epoch Definition

In this section, we clarify the notion of "epoch" in our previous experiments, where an epoch means a full pass over the validation set (for upper-level variable x update). For a more comprehensive comparison, we re-conduct experiments where

an epoch means a full pass over the training set (for lower-level variable y update). In this case there are fewer x updates than the previous run due to the warm-start phase. The results show that SLIP is still empirically better than other baselines.

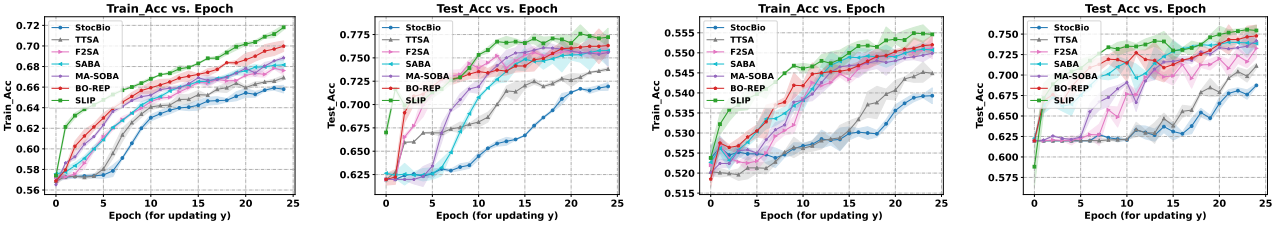
G.1. Hyper-representation Learning



(a) Training accuracy on SNLI (b) Test accuracy on SNLI (c) Training accuracy on ARD (d) Test accuracy on ARD

Figure 4. Comparison with bilevel optimization baselines on Hyper-representation. Figure (a) and (b) are the results in the SNLI dataset. Figures (c) and (d) are the results of the Amazon Review Dataset (ARD).

G.2. Data Hyper-cleaning



(a) Training acc with $p = 0.2$ (b) Test acc with $p = 0.2$ (c) Training acc with $p = 0.4$ (d) Test acc with $p = 0.4$

Figure 5. Comparison with bilevel optimization baselinses on data hyper-cleaning. Figure (a), (b) are the results with the corruption rate $p = 0.2$. Figure (c), (d) are the results with the corruption rate $p = 0.4$.