2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

A Loopless Distributed Algorithm for Personalized Bilevel Optimization

Youcheng Niu[†], Ying Sun[‡], Yan Huang[†] and Jinming Xu[†]

***Abstract*— This paper studies a class of personalized distributed bilevel optimization problems over networks, where nodes aim at jointly optimizing the sum of outer-level objectives that depend on the solution of inner-level optimization problems. The existing algorithms for distributed bilevel optimization problems usually require extra computation loops for estimating hypergradients. To facilitate computational efficiency, we develop a loopless distributed algorithm that employs certain steps to approximate the optimal solution of inner-level optimization problems, and track Hessian-inverse-vector products in a recursive manner. We prove that for stochastic nonconvex-strongly-convex problems, the proposed algorithm achieves the state of the art $\mathcal{O}(\epsilon^{-2})$ communication cost, while improving the computational cost by $\mathcal{O}(\log(\frac{1}{\epsilon}))$. Numerical experiments validate our theoretical findings.**

I. Introduction

Bilevel optimization (BO) problems have recently received increasing attentions, which generally takes the form of $\min_{x\in\mathbb{R}^n} \Phi(x) = f(x,\theta^*(x))$, s.t. $\theta^*(x) = \arg\min_{\theta\in\mathbb{R}^p} g(x,\theta)$ where f and g are outer- and inner-level functions, respectively. Such bilevel structures provide a favorable framework in formulating some important applications, ranging from compositional optimization problems and Stackelberg game [1], to modern machine learning problems such as meta-learning [2], reinforcement learning [3], hyperparameter optimization [3], to name a few. For examples, in meta-learning [2], the outer-level function can be utilized to capture the aggregate features of multiple tasks or datasets, while the inner-level function is used to adapt to new tasks with limited samples. Recently, with the increasing amount of data and growing concerns regarding privacy issues, investigating bilevel optimization in distributed settings is also highlighted, where multiple nodes aim to collectively optimize a common global model, while each node maintain a local copy of the global model and exchange the local update with its neighbors for consensus. However, when it comes to multiple tasks or statistically heterogeneous datasets, the single shared model may not perform well in distributed settings [4].

Inspired by the above facts, this paper considers a class of personalized distributed bilevel optimization (PDBO) that accounts for the personality of local tasks while taking into account the shared characteristics across all tasks. In particular, in the PDBO problem, there is a network of m nodes collaborating to solve the following problem:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} \Phi(x) = \frac{1}{m}\sum_{i=1}^{m} f_i\left(x,\theta_i^*(x)\right) \\ &\text{s.t. } \theta_i^*(x) = \arg\min_{\theta_i\in\mathbb{R}^p} g_i(x,\theta_i), \end{aligned} \tag{1}$$

where x and θ_i are the global and local personalized decision variables, respectively, while $f_i(x,\theta) = \mathbb{E}_{\varsigma_i\sim\mathcal{D}_{f_i}}[\hat{f}_i(x,\theta,\varsigma_i)]$ is the outer-level overall (possibly nonconvex) loss function and $g_i(x,\theta) = \mathbb{E}_{\xi_i\sim\mathcal{D}_{g_i}}[\hat{g}_i(x,\theta,\xi_i)]$ is the inner-level personalized objective which is assumed to be strongly convex in θ for all x. In distributed settings, each node i only has access to its local functions f_i and g_i, and is allowed to interact with its neighbors over a peer-to-peer network.

Bilevel optimization differs from general constrained optimization in that outer-level functions depend on the solutions of inner-level optimization problems. This feature makes it challenging to estimate the gradients of outer-level functions (i.e., hypergradients). A variety of approximation methods have been proposed towards addressing the above challenge, which can be generally categorized into two main streams: iterative differentiation (ITD) and approximate implicit differentiation (AID) methods. AID-based methods involve two sub-problems at each iteration, one for computing Hessian inverse matrices and the other for determining near-optimal solutions of the inner-level functions. Typical approaches to approximating the Hessian inverse matrices include Neumann Series (NS) approaches (e.g., BSA [5], TTSA [3]) and conjugate gradient descent (CG) approaches (e.g., STABLE [6], stoBiO [7]), all of which require an inner loop of computing process. Hence, for nonconvex-strongly-convex BO problems, these algorithms [5]–[7] are able to reach a computational complexity of the order $\mathcal{O}(\epsilon^{-2}\log\frac{1}{\epsilon})$. By modifying the CG, NS and ITD methods, a few works [8]–[10] further put forward centralized BO algorithms with a computational complexity of $\mathcal{O}(\epsilon^{-2})$.

All of the abovemetioned algorithms focus on centralized BO problems. However, distributed BO problems also arise in practical application domains where tasks are large-scale and datasets are scatteredly stored. This above feature renders existing single-level distributed algorithms invalid, such as distributed gradient descent (DGD) [11], [12], dual multiplier [13], and gradient tracking (GT) [14], [15], due to the absence of explicit knowledge of optimal solutions to the inner-level problems. Recently, a few works [16], [17], [18] have been developed to solve a class of global DBO problems that formulate both of the inner-level and outer-level objectives

The work of Niu, Huang and Xu has been supported in parts by National Natural Science Foundation of China under Grants 62003302, 62373323 and 62088101. The work of Sun was partially supported by the Office of Naval Research under the Grant N00014-21-1-2673.

[†]Youcheng Niu, Yan Huang and Jinming Xu are with the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China. Correspondence to jimmyxu@zju.edu.cn (Jinming Xu).

[‡]Ying Sun is with School of Electrical Engineering and Computer Science, The Pennsylvania State University, State College, PA 16802.

as a finite sum. For examples, Chen *et al* [16] propose a double-loop distributed BO algorithm based on CG and DGD methods to solve global DBO problem. By employing NS methods and approximating the inner-level solutions, single-loop algorithms are developed in [17], [18] but both require extra computation loops for estimating Hessian inverse matrices. It should be noted that the global DBO problem considered in [16], [17], [18] are all using a single shared model, which may do not perform well for node-specific tasks or statistically heterogeneous data subsets. To overcome this issue, Lu *et al* [4] investigate the PDBO problem and develop a stochastic primal-dual decentralized algorithm (SPDB) by employing the same approximation technique used in the BSA algorithm [5] for estimating Hessian inverse matrices. It is worth noting that, however, all of these existing distributed algorithms [4], [16], [17] requires extra computation loops for estimating either Hessian inverse matrices or near-optimal solutions to inner-level problems, leading to a computational complexity $\mathcal{O}(\epsilon^{-2}\log(\frac{1}{\epsilon}))$. Although some works employ the variance reduction techniques to improve this complexity to $\mathcal{O}(\epsilon^{-\frac{3}{2}}\log(\frac{1}{\epsilon}))$ [18], the extra computation loops are still not avoided, inducing relatively high computation cost. As a result, it is important to explore a computationally effective distributed algorithm for the PDBO problem with lower computational complexity.

Main Contributions of This Work: This paper proposes a new loopless distributed algorithm (termed L-PDBO) for the PDBO problem (1). The L-PDBO algorithm does not involve extra computation loops and thus enjoys computational advantages. Specifically, in estimating the hypergradients, the L-PDBO algorithm employs one-step gradient descent to approximate the near-optimal solution of the personalized variables at each iteration. More importantly, the L-PDBO algorithm introduces auxiliary variables to track Hessian-inverse-vector products in a recursive manner. Theoretically, it is shown that the L-PDBO algorithm can converge to a ϵ-stationary point with respective communication and computational complexity of $\mathcal{O}(\epsilon^{-2})$ for the DPBO problem (1). This computational complexity outperforms those of existing state-of-the-art distributed bilevel optimization algorithms by the order of $\mathcal{O}(\log(\frac{1}{\epsilon}))$. Finally, numerical experiments further demonstrate the superiority of the proposed algorithm on computation complexity and running efficiency.

II. Algorithm Development

In this paper, we consider an undirected and connected network modeled as a graph $\mathcal{G}=\{\mathcal{V},\mathcal{E}\}$ with $\mathcal{V}$ denoting the node set and $\mathcal{E}\subset V\times V$ denoting the edge set. For the DPBO problem (1), we aim at finding a solution $(x^*,\ \theta_1^*,\cdots,\ \theta_m^*)$ that satisfies the following first-order stationary condition:

$$\nabla\Phi\left(x^*\right)=0, \nabla_\theta g_i(x^*,\theta_i^*)=0,\ \forall i\in\{1,\cdots,m\}. \quad (2)$$

A general distributed algorithm for seeking (2) is based on gradient descent methods (GD) and average consensus strategies. To be more specific, we let $\mathcal{N}_i$ denote neighbor set of node i, i.e., $j\in\mathcal{N}_i$ if $(i,j)\in\mathcal{E}$. At each iteration $k+1$, each node i introduces a local estimate x_i for the global variable x, and then performs the following update:

$$x_i^{k+1}=\sum\nolimits_{j\in\mathcal{N}_i} w_{ij}x_j^k-\alpha\nabla f_i\left(x_i^k,\theta_i^*(x_i^k)\right), \quad (3)$$

where $\alpha>0$ is a step-size and $W=[w_{ij}]$ is a weight matrix for reaching consensus on local estimates of nodes, while $\nabla f_i\left(x_i^k,\theta_i^*(x_i^k)\right)$ is the local hypergradient whose close-form expression can be derived based on the implicit differentiation methods [5] as follows:

$$\begin{aligned}\nabla f_i\left(x_i^k,\theta_i^*(x_i^k)\right)=&\nabla_x f_i\left(x_i^k,\theta_i^*(x_i^k)\right)\\&+\nabla\theta_i^*(x_i^k)\nabla_\theta f_i(x_i^k,\theta_i^*(x_i^k))\ ,\end{aligned} \quad (4)$$

where $\nabla\theta_i^*(x_i^k)=-\nabla_{x\theta}^2 g_i(x_i^k,\theta_i^*(x_i^k))\big[\nabla_{\theta\theta}^2 g_i(x_i^k,\theta_i^*(x_i^k))\big]^{-1}$, and $\nabla_{x\theta}^2 g_i$ and $\nabla_{\theta\theta}^2 g_i$ respectively denote Jacobian and Hessian matrices, while $\nabla_x f_i$ and $\nabla_\theta f_i$ denote the partial gradients. Implementing directly the distributed GD method given in (3) to solve the DPBO problem (1) is computationally expensive. To be precise, computing the hypergradient (4) requires the knowledge of the personalized solutions $\theta_i^*(x_i^k)$ and the Hessian inverse matrices $\left[\nabla_{\theta\theta}^2 g_i(x_i^k,\theta_i^*(x_i^k))\right]^{-1}$, whose calculations involve an iterative procedure. To address this issue and reduce the computation cost, we adopt the following one-step approximation strategies (S1) and (S2):

(S1) One-Step Approximation for $\theta_i^*(x_i^k)$: We know that obtaining $\theta_i^*(x_i^k)$ will require solving the strongly convex optimization problem $\min_{\theta_i}\ g_i(x_i^k,\theta_i)$. To eliminate the need for such computation loop, we perform one-step gradient descent (6b) with the step-size $\beta>0$ for the personalized variables θ_i at each iteration, where we reuse their previous values to compute the gradients for a better approximation of $\theta_i^*(x_i^k)$ and employ the stochastic estimator $\nabla_\theta\hat{g}_i$ with a sample $\xi_{i,1}^{k+1}$ for $\nabla_\theta g_i$. In this way, replacing $\theta_i^*(x_i^k)$ by θ_i^k we obtain a surrogate of the hypergradients in the form:

$$\begin{aligned}\bar{\nabla}f_i\left(x_i^k,\theta_i^k\right)&=\nabla_x\hat{f}_i\left(x_i^k,\theta_i^k;\varsigma_{i,2}^{k+1}\right)\\&-\nabla_{x\theta}^2\hat{g}_i(x_i^k,\theta_i^k,\xi_{i,3}^{k+1})\big[\nabla_{\theta\theta}^2 g_i(x_i^k,\theta_i^k)\big]^{-1}\nabla_\theta\hat{f}_i(x_i^k,\theta_i^k),\end{aligned} \quad (5)$$

where the stochastic estimators $\nabla_x\hat{f}_i$ and $\nabla_{x\theta}\hat{g}_i$ with samples $\varsigma_{i,2}^{k+1}$ and $\xi_{i,3}^{k+1}$ are used respectively for $\nabla_x f_i$ and $\nabla_{x\theta}^2 g_i$. By employing this strategy, it is expected that $\bar{\nabla}f_i\left(x_i^k,\theta_i^k\right)$ will converge to $\nabla f_i\left(x_i^k,\theta_i^*(x_i^k)\right)$ as θ_i^k approximates its optimal counterpart $\theta_i^*(x_i^k)$.

(S2) One-Step Approximation for $\left[\nabla_{\theta\theta}^2 g_i(x_i^k,\theta_i^k)\right]^{-1}$: To avoid evaluating directly the Hessian inverse, we group it together with $\nabla_\theta f(x_i^k,\theta_i^k)$ and consider evaluating approximately the Hessian-inverse-vector products $\left[\nabla_{\theta\theta}^2 g_i(x_i^k,\theta_i^k)\right]^{-1}\nabla_\theta f(x_i^k,\theta_i^k)$ with auxiliary variables v_i^k. The natural way is to employ conjugate gradient (CG) methods. In order to avoid producing a computation loop, different from [4], [16], [17], [18], we opt to update v_i^{k+1} only once at each iteration, warm started using their values at the previous iteration. In this way, we design the update rule (6c) for v_i^{k+1}, where $\lambda>0$ is the step-size and the stochastic estimators $\nabla_{\theta\theta}^2\hat{g}_i$ and $\nabla_\theta\hat{f}_i$ with samples $\xi_{i,2}^{k+1}$ and $\varsigma_{i,1}^{k+1}$ are considered respectively for $\nabla_{\theta\theta}^2 g_i$ and $\nabla_\theta f_i$.

Combining the above two approximation strategies and tak-

Algorithm 1 L-PDBO

1: **Require**: Set $x_i^0 = \tilde{x}$ for $i \in \mathcal{V}$ with arbitrary $\tilde{x} \in \mathbb{R}^n$, and initialize θ_i^0, s_k^0, v_i^0 as well as $\{\alpha, \beta, \lambda\}$.

2: **for** $k = 0, 1, 2..., K$, each node $i \in \mathcal{V}$ in parallel **do**

3: **Sample** $\xi_{i,1}^{k+1}, \xi_{i,2}^{k+1}, \xi_{i,3}^{k+1}, \varsigma_{i,1}^{k+1}, \varsigma_{i,2}^{k+1}$.

4: **Communicate with neighboring node** $j \in \mathcal{N}_i$.

5: **Update the outer-level variables**:

$$x_i^{k+1} = \sum\nolimits_{j \in \mathcal{N}_i} w_{ij} x_j^k - \alpha s_i^k; \tag{6a}$$

6: **Update the inner-level variables**:

$$\theta_i^{k+1} = \theta_i^k - \beta \nabla_\theta \hat{g}_i(x_i^k, \theta_i^k; \xi_{i,1}^{k+1}); \tag{6b}$$

7: **Update the auxiliary estimates of the Hessian-inverse-vector products**:

$$\begin{aligned} v_i^{k+1} = &(I - \lambda \nabla_{\theta\theta}^2 \hat{g}_i(x_i^k, \theta_i^k; \xi_{i,2}^{k+1}))v_i^k \\ &+ \lambda \nabla_\theta \hat{f}_i(x_i^k, \theta_i^k; \varsigma_{i,1}^{k+1}); \end{aligned} \tag{6c}$$

8: **Approximate the hypergradients**:

$$\begin{aligned} s_i^{k+1} = &\nabla_x \hat{f}_i(x_i^{k+1}, \theta_i^{k+1}; \varsigma_{i,2}^{k+1}) \\ &- \nabla_{x\theta}^2 \hat{g}_i(x_i^{k+1}, \theta_i^{k+1}; \xi_{i,3}^{k+1}) v_i^{k+1}. \end{aligned} \tag{6d}$$

9: **end for**

ing into account the stochasticity in gradient and Hessian evaluation, we propose a loopless distributed algorithm for the personalized bilevel optimization problem (1) (termed L-PDBO), described in Algorithm 1. In Algorithm 1, it is assumed that each node can query locally independent samples $\xi_{i,1}^{k+1}, \xi_{i,2}^{k+1}, \xi_{i,3}^{k+1}, \varsigma_{i,1}^{k+1}, \varsigma_{i,2}^{k+1}$ and utilize the mini-batch gradients to perform the updates.

III. Convergence Analysis

In this section, we will examine the convergence of the proposed L-PDBO algorithm in stochastic nonconvex-strongly-convex cases. In order to analyze the convergence of the L-PDBO algorithm, we begin by making several standard assumptions that are commonly adopted in the literature on bilevel optimization. We also provide some key properties that will aid in our subsequent analysis.

A. Standard Assumptions

The following four assumptions mainly concern the continuity of the outer- and inner-level functions, stochasticity of the gradient estimates and network connectivity.

*Assumption 1 (**Outer-level functions**):* Each outer-level function f_i, $i \in \mathcal{V}$ satisfies the following properties:

i) $f_i : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$ is jointly differentiable;

ii) $\nabla_x f_i(x, \theta)$ is $L_{f,x}$-Lipschitz-continuous w.r.t x uniformly for $\theta \in \mathbb{R}^p$, and $\nabla_\theta f_i(x, \theta)$ is $L_{f,\theta}$-Lipschitz-continuous w.r.t θ uniformly for $x \in \mathbb{R}^n$;

iii) $\nabla_x f_i(x, \theta)$ and $\nabla_\theta f_i(x, \theta)$ are bounded with positive constants $C_{f,x}$ and $C_{f,\theta}$, respectively;

iv) $\nabla_{\theta\theta}^2 f_i(x, \theta)$ and $\nabla_{\theta x}^2 f_i(x, \theta)$ are $L_{f,\theta\theta}$- and $L_{f,\theta x}$-Lipschitz-continuous in $x \in \mathbb{R}^n$ and $\theta \in \mathbb{R}^p$, respectively.

*Assumption 2 (**Inner-level functions**):* Each inner-level function g_i, $i \in \mathcal{V}$ satisfies the following properties:

i) $g_i(x, \theta)$ is μ_g-strongly convex w.r.t θ given any $x \in \mathbb{R}^n$ and is three times continuously differentiable for all $x \in \mathbb{R}^n$ and $\theta \in \mathbb{R}^p$;

ii) $\nabla_\theta g_i(x, \theta)$ is $L_{g,\theta}$-Lipschitz-continuous w.r.t. θ uniformly for all $x \in \mathbb{R}^n$, and $\nabla_{x\theta}^2 g_i(x, \theta)$, $\nabla_{\theta\theta}^2 g_i(x, \theta)$ are jointly $L_{g,x\theta}$- and $L_{g,\theta\theta}$-Lipschitz-continuous in $x \in \mathbb{R}^n$ and $\theta \in \mathbb{R}^p$, respectively;

iii) $\nabla_{x\theta}^2 g_i(x, \theta)$ is bounded with positive constant $C_{g,\theta x}$;

iv) $\nabla_{\theta\theta x}^3 g_i(x, \theta)$ is $L_{g,\theta\theta x}$-Lipschitz-continuous in $x \in \mathbb{R}^n$ and $\theta \in \mathbb{R}^p$.

Next, we provide the assumptions on the stochastic gradient estimates and networks involved in Algorithm 1.

*Assumption 3 (**Stochastic gradient estimates**):* The samples $\xi_{i,1}^{k+1}$, $\xi_{i,2}^{k+1}$, $\xi_{i,3}^{k+1}$, $\varsigma_{i,1}^{k+1}$, $\varsigma_{i,2}^{k+1}$ are independent for any $k+1$. Moreover, $\nabla_\theta \hat{g}_i(x, \theta; \xi_{i,1}^{k+1})$, $\nabla_{\theta\theta}^2 \hat{g}_i(x, \theta; \xi_{i,2}^{k+1})$, $\nabla_{x\theta}^2 \hat{g}_i(x, \theta; \xi_{i,3}^{k+1})$, $\nabla_\theta \hat{f}_i(x, \theta; \varsigma_{i,1}^{k+1})$, $\nabla_x \hat{f}_i(x, \theta; \varsigma_{i,2}^{k+1})$ are respectively unbiased estimators of $\nabla_\theta g_i(x, \theta)$, $\nabla_{\theta\theta}^2 g_i(x, \theta)$, $\nabla_{x\theta}^2 g_i(x, \theta)$, $\nabla_\theta f_i(x, \theta)$, $\nabla_x f_i(x, \theta)$ and their variances are bounded by σ^2.

*Assumption 4 (**Connectivity of network**):* The communication network $\mathcal{G}$ is connected, and the weight matrix $W = [w_{ij}]_{i,j=1}^m$ associated with the network $\mathcal{G}$ with $w_{ij} > 0$ is doubly stochastic such that there exists a constant $\rho = \|W - J\|^2 \in [0, 1)$ with $J = \frac{1_m 1_m^{\mathrm{T}}}{m}$.

Assumptions 1-3 are commonly used in stochastic bilevel optimization literature [3], [6], [10], [18]–[20]. In particular, inspired by [10], [21], Assumptions 1(iv) and 2(iv) are considered to ensure the smoothness of the Hessian-inverse-vector products and provide a tighter analysis in stochastic cases. Assumption 4 on the weighted matrix is general in the distributed optimization literature to ensure the convergence of distributed algorithms [14].

B. Technical Preliminaries

In what follows, we provide some technical preliminaries that will facilitate subsequent convergence analysis.

*Proposition 1 (**Inner solutions** [5], [21]):* Suppose Assumptions 1-3 hold. Then $\theta_i^*(x) = \arg\min_{\theta_i} g_i(x, \theta_i)$ and $\nabla\theta_i^*(x)$ are Lipschitz-continuous w.r.t. x with parameters L_{θ^*} and $L_{\theta^*,x}$, respectively.

*Proposition 2 (**Hypergradients** [3], [5]):* Suppose Assumptions 1-3 hold. Then $\nabla\Phi(x)$ is Lipschitz-continuous with parameter L.

*Proposition 3 (**Hessian-inverse-vector products** [5], [10]):* Suppose Assumptions 1-3 hold. Letting $v_{\theta_i^*(x)} = \left[\nabla_{\theta\theta}^2 g_i(x, \theta_i^*(x))\right]^{-1} \nabla_x f_i(x, \theta_i^*(x))$ for $i \in \mathcal{V}$, then $v_{\theta_i^*(x)}$ and $\nabla v_{\theta_i^*(x)}$ are Lipschitz-continuous w.r.t x with parameters L_v and $L_{v,x}$, respectively.

Propositions 1-3 respectively depict the hidden smoothness of $\theta_i^*(x)$, $\Phi(x)$, $v_{\theta_i^*}$. Next, we show the boundness of the auxiliary estimates for the Hessian-inverse-vector products.

*Proposition 4 (**Auxiliary estimates**):* Suppose Assumptions 1-3 hold. Then we have that $\|v_{\theta_i^*}\| \le M$ and $\|v_i^k\| \le M$ with parameter $M = \frac{C_{f,\theta}}{\mu_g}$, for any k.

Propositions 1-3 can be derived based on Assumptions 1-3 and Proposition 4 can be obtained by induction arguments.

The proof and specific expressions of related Lispchitz parameters are omitted here due to space limitations. With the standard assumptions and auxiliary results, we are now ready to present the following main theoretical results.

C. Main Results

In what follows, we use $\mathcal{F}^k = \sigma\left\{\bigcup_{i=1}^m x_i^k, \theta_i^k, v_i^k, \cdots, x_i^0, \theta_i^0, v_i^0\right\}$ to denote the filtration up to iteration k, and letters with a bar the average, e.g., $\bar{x} = (1/m)\sum_{i=1}^m x_i$. We start by showing the descent of Φ along the averaged iterates $\{\bar{x}^k\}$.

*Lemma 1 (**Descent lemma**):* Consider the sequence $\{x_i^k, s_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-4 hold. Then, we have

$$\begin{aligned}&\mathbb{E}\left[\Phi(\bar{x}^{k+1})\right]\\ \leq&\mathbb{E}[\Phi(\bar{x}^k)] - \frac{\alpha}{2}\mathbb{E}[\|\nabla\Phi(\bar{x}^k)\|^2] - \frac{\alpha}{2}(1-\alpha L)\,\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2]\\ &+\frac{\alpha}{2}\mathbb{E}[\|\nabla\Phi(\bar{x}^k) - \mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2] + \frac{\alpha^2 L}{2}\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k] - \bar{s}^k\|^2].\end{aligned} \tag{10}$$

Proof: See Appendix VI-A. ∎

The above descent lemma indicates that, the descent of the overall objective functions depends on the hypergradient approximation errors as well as the stochastic errors. Therefore, we proceed to establish the upper bounds for the terms $\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k] - \bar{s}^k\|^2]$ and $\mathbb{E}[\|\nabla\Phi(\bar{x}^k) - \mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2]$.

*Lemma 2 (**Hypergradient variances**):* Consider the sequence $\{x_i^k, \theta_i^k, v_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-3 hold. Then, we have

$$\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k] - \bar{s}^k\|^2] \leq \frac{1}{m}(\sigma^2 + M^2\sigma^2), \tag{11}$$

and

$$\begin{aligned}&\mathbb{E}[\|\nabla\Phi(\bar{x}^k) - \mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2]\\ \leq&D\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|x_i^k - \bar{x}^k\|^2] + D\frac{1}{m}\sum_{i=1}^m\mathbb{E}\|\theta_i^k - \theta_i^*(\bar{x}^k)\|^2]\\ &+4C_{g,x\theta}^2\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_i^k - v_{\theta_i^*(\bar{x}^k)}\|^2],\end{aligned} \tag{12}$$

where $D \triangleq 2L_{f,x}^2 + 4M^2L_{g,x\theta}^2$.

Proof: See Appendix VI-B. ∎

Furthermore, the following two lemmas show the contraction properties of the last two terms of (12).

*Lemma 3 (**Hessian-inverse-vector product errors**):* Consider the sequence $\{x_i^k, \theta_i^k, v_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-4 hold. Let $c_\lambda = \lambda/\alpha$ denote the ratio of step-size λ and α. If α and λ respectively satisfy $\alpha < \frac{c_\lambda\mu_g}{4\left(c_\lambda^2L_{g,\theta}^2+U^2\right)}$ and $\lambda < \frac{1}{\mu_g}$ with $U \triangleq 2(C_{f,x}^2 + C_{g,x\theta}^2M) + \left(1+M^2\right)\sigma^2$, then we have

$$\begin{aligned}&\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_i^{k+1} - v_{\theta_i^*(\bar{x}^{k+1})}\|^2]\\ \leq&(1-\frac{\lambda\mu_g}{4})\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_i^k - v_{\theta_i^*(\bar{x}^k)}\|^2]\\ &+q_x\alpha\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|x_i^k - \bar{x}^k\|^2 + \|\theta_i^k - \theta_i^*(\bar{x}^k)\|^2]\\ &+q_s\alpha^2\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2] + q_\sigma\alpha^2\sigma^2,\end{aligned} \tag{13}$$

where $q_x \triangleq (\frac{3B}{\mu_g} + M^2L_{g,\theta\theta}^2\lambda + L_{f,\theta}^2\lambda)c_\lambda$, $q_\sigma \triangleq \left(1+M^2\right)(4L_v^2 + L_{v,x}^2) + \left(1+M^2\right)c_\lambda^2$, $q_s \triangleq \frac{((4+a_1)\lambda\varpi+1)L_v^2}{\lambda\varpi}$ with $B \triangleq 2L_{g,\theta\theta}^2M^2 + L_{f,\theta}^2$, $\varpi \triangleq \frac{\mu_g}{2}$ and $a_1 \triangleq \frac{L_{v,x}^2}{L_v^2}$.

Proof: See Appendix VI-C. ∎

*Lemma 4 (**Inner-level errors**):* Consider the sequence $\{x_i^k, \theta_i^k, v_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-3 hold. Let $c_\beta = \beta/\alpha$ denote the ratio of the step-sizes β and α. If α and β respectively satisfy $\alpha < \frac{c_\beta\omega}{2\left(c_\beta^2L_{g,\theta}^2+U^2\right)}$ and $\beta < b \triangleq \min\left\{\frac{2}{\mu_g+L_{g,\theta}}, \frac{\mu_g+L_{g,\theta}}{2\mu_gL_{g,\theta}}\right\}$, then we have:

$$\begin{aligned}&\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|\theta_i^{k+1} - \theta_i^*(\bar{x}^{k+1})\|^2]\\ \leq&(1-\beta\frac{\mu_gL_{g,\theta}}{4(\mu_g+L_{g,\theta})})\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|\theta_i^k - \theta_i^*(\bar{x}^k)\|^2]\\ &+p_x\alpha\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|x_i^k - \bar{x}^k\|^2] + p_s\alpha^2\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2] + p_\sigma\beta^2\sigma^2,\end{aligned} \tag{14}$$

where $p_x \triangleq (2\beta + \frac{\mu_g+L_{g,\theta}}{\mu_gL_{g,\theta}})c_\beta L_{g,\theta}^2$, $p_\sigma \triangleq (1+M^2)(2L_{\theta^*}^2 + L_{\theta^*,x}^2) + c_\beta^2$, $p_s \triangleq \frac{((2+a_2)\omega\beta+1)L_{\theta^*}^2}{\omega\beta}$ with $\omega \triangleq \frac{\mu_gL_{g,\theta}}{2(\mu_g+L_{g,\theta})}$ and $a_2 \triangleq \frac{L_{\theta^*,x}^2}{L_{\theta^*}^2}$.

Proof: See Appendix VI-D. ∎

To control the Hessian-inverse-vector product errors and inner-level errors, we also need to examine the consensus errors that emerge from the sparse network structure. To this end, we proceed to establish the convergence properties of the consensus errors.

*Lemma 5 (**Consensus errors**):* Consider the sequence $\{x_i^k, \theta_i^k, v_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-4 hold. Then, we have for $1 \leq k \leq K$

$$\sum_{i=1}^m\mathbb{E}[\|x_i^{k+1} - \bar{x}^{k+1}\|^2] \leq \frac{4mU\alpha^2}{(1-\rho)^2}, \tag{15}$$

where $\rho = \|(W-J)\otimes I_n\|^2 \in [0,1)$.

Proof: See Appendix VI-E. ∎

In conjunction with the convergence properties of the consensus errors, Lemma 3 and 4 suggest that, the estimates θ_i^k and v_i^k generated by the approximation strategies (6b)-(6d) will asymptotically converge to the optimal personalized variables and the Hessian-inverse-vector products evaluated at $\bar{x}^k$, respectively. In deterministic cases, we can also employ Young's inequality to control the term Z in the equality (23) and deal with Hessian-inverse-vector product

errors without Assumptions 1(iv) and 2(iv).

Next, before presenting our convergence results, we introduce a potential function as follows:

$$\Delta^k = c_\lambda \mu_g \Xi \Phi(\bar{x}^k) + 2C_{g,x\theta}^2 \Xi \frac{1}{m} \sum_{i=1}^m \left\| v_i^k - v_{\theta_i^*(\bar{x}^k)} \right\|^2 + \Gamma \frac{1}{m} \sum_{i=1}^m \left\| \theta_i^k - \theta_i^*(\bar{x}^k) \right\|^2, \tag{16}$$

where $\Xi \triangleq \frac{c_\beta \mu_g L_{g,\theta}}{4(\mu_g + L_{g,\theta})}$ and $\Gamma \triangleq \frac{c_\lambda \mu_g D}{2} + 8C_{g,x\theta}^2 q_x$.

Theorem 1: Consider the sequence $\{x_i^k, \theta_i^k, v_i^k\}$ generated by Algorithm 1. Suppose Assumptions 1-3 hold. If $\alpha = \min\left(u, \left(\frac{r_0}{r_1 \sigma^2 (K+1)}\right)^{\frac{1}{2}}, \left(\frac{r_0 (1-\rho)^2}{r_2 (K+1)}\right)^{\frac{1}{3}}\right)$, where the specific definition of r_0, r_1, r_2 can be found in Appendix, with the positive constant u satisfying

$$u < \min\left\{ \frac{1}{2L}, \frac{c_\lambda \mu_g}{4\left(c_\lambda^2 L_{g,\theta}^2 + U^2\right)}, \frac{c_\beta \omega}{2\left(c_\beta^2 L_{g,\theta}^2 + U^2\right)}, \frac{1}{c_\beta b}, \frac{1}{2C_{g,x\theta}(4L_v + L_{v,x})}, \frac{1}{2(2L_{\theta^*} + L_{\theta^*,x})\sqrt{D + B\frac{16C_{g,x\theta}^2}{\mu_g \mu_g}}}, \frac{1}{c_\lambda \mu_g} \right\} \tag{17}$$

for some positive constants c_β and c_λ; and $\beta = c_\beta \alpha$, $\lambda = c_\lambda \alpha$, then we have

$$\frac{1}{K+1} \sum_{k=0}^K \mathbb{E}[\|\nabla \Phi(\bar{x}^k)\|^2] \leq \mathcal{O}\left(\frac{1}{u(K+1)} + \frac{\sigma}{(K+1)^{1/2}} + \frac{1}{(1-\rho)^{2/3}(K+1)^{2/3}} \right). \tag{18}$$

Proof: See Appendix VI-F. ■

Remark 1: Theorem 1 indicates that the L-PDBO algorithm has a convergence rate of $\mathcal{O}(\frac{1}{\sqrt{K}})$, which matches those of state-of-the-art distributed algorithms for the PDBO problem in stochastic nonconvex-strongly-convex cases. In particular, if the step-size $\alpha = \frac{u}{\sqrt{K+1}}$, we can derive a rate of $\frac{1}{K+1}\sum_{k=0}^K \mathbb{E}[\|\nabla \Phi(\bar{x}^k)\|^2] \leq \mathcal{O}\left(\frac{1}{u(K+1)^{1/2}} + \frac{\sigma^2}{(K+1)^{1/2}} + \frac{1}{(1-\rho)^2(K+1)}\right)$. In both cases, the convergence rate is affected by the consensus errors and stochastic errors, in which the former decays faster than the latter. Thus, as the number of iterations increases, the algorithm's convergence becomes increasingly independent of the network structure, and the impact of the stochastic error will become predominant. Thanks to the loopless structure that avoids the extra computation loops, both the communication and computation (gradient evaluation) complexity of the proposed L-PDBO algorithm to reach a ϵ-stationary point are in an order of $\mathcal{O}(\epsilon^{-2})$. It should be noted that this is in contrast to the existing works [4], [16], [20], which require a total number of $\mathcal{O}(\epsilon^{-2}\log\frac{1}{\epsilon})$ gradient evaluation due to the extra computation loops for estimating the near-optimal inner solutions or Hessian-inverse-vector products. Thus, the proposed L-PDBO algorithm improves the computational complexity by the order of $\mathcal{O}(\log\frac{1}{\epsilon})$.

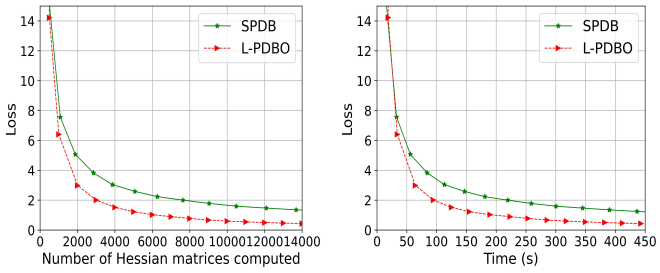

Fig. 1: Comparison of the proposed algorithm and the SPDB algorithms in terms of training loss.

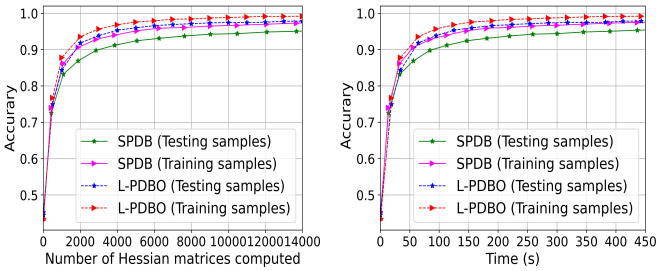

Fig. 2: Comparison of the proposed algorithm and the SPDB algorithm in terms of training and testing accuracy.

TABLE I: Number of Hessian matrices needed to be computed to reach a given testing accuracy.

Accuracy	0.65	0.7	0.8	0.9	0.95
NHMC of SPDB	320	410	920	3000	9530
NHMC of L-PDBO	330	360	690	1650	3700

NHMC: Number of Hessian matrices computed.

IV. Numerical Experiments

In this section, the numerical experiments are provided to validate the performance of the L-PDBO algorithm. Specifically, we investigate a distributed hyper-parameter optimization on logistic regression problem with MNIST dataset. Here, a binary classification task is considered on a dataset of 4000 samples (s_{ij}, b_{ij}) consisting of the digit '0' and the digit '1', where s_{ij} represents the ij-th feature in $\mathbb{R}^{784}$ and b_{ij} is the corresponding label. Furthermore, the outer- and inner-level functions in the problem (1) respectively take form of $f_i(\lambda, \theta_i^*(\lambda)) = \sum_{(b_{ij}, s_{ij}) \in D_i^{\text{val}}} \phi_i(b_{ij} s_{ij}^{\mathrm{T}} \theta_i^*(\lambda))$ and $g_i(\lambda, \theta_i) = \sum_{(b_{ij}, s_{ij}) \in D_i^{\text{train}}} \phi_i(b_{ij} s_{ij}^{\mathrm{T}} \theta_i) + R(\lambda, \theta_i)$, with a loss function $\phi_i(x) = \log(1 + e^{-x})$ and a regularizer $R(\lambda, \theta_i) = \theta_i^{\mathrm{T}} \mathrm{diag}(e^\lambda) \theta_i$. For this problem, whole nodes aim to jointly seek the best hyperparameter λ utilizing their optimal personalized models $\theta_i^*(\lambda)$ that are trained by their private training dataset D_i^{train}. In this experiment, a connected network with 10 nodes and an edge connectivity of 0.5 is used. Each node i is assigned with 400 samples randomly with 200 samples reserved for the validation set

D_i^{val} and the remaining 200 samples used for the training set D_i^{train}. The step-sizes of the L-PDBO algorithm are set as: $\alpha = 0.01$, $\beta = 0.016$, $\lambda = 0.01$. We compare it with the SPDB algorithm [4] that is based on the Neumann series approaches with a sampling scheme of $\lceil \frac{1}{2}\log(\sqrt{k+1})\rceil$ [3] for estimating Hessian inverse matrices.

The experiment results are provided in Figs. 1 and 2 and Table I. It can be seen from Fig. 1 that the L-PDBO algorithm enjoys faster convergence with respect to the running time and the number of Hessian matrices computed, compared to the state-of-the-art SPDB algorithm. It follows from Fig. 2 that the L-PDBO algorithm is able to achieve a desirable testing and training accuracy in less time. Table I further reveals the lower complexity of the L-PDBO algorithm in terms of the number of Hessian matrices computed. These results demonstrate the superiority of the L-PDBO algorithm in computational complexity. This superiority will become even more pronounced when the computation of the Hessian matrices is computationally expensive.

V. Conclusions

This paper proposed a loopless distributed algorithm to solve the PDBO problem by employing two key approximation steps for estimating the inner-level optimal solution as well as the hypergradients without involving extra computation loops. It was theoretically proved that, the proposed algorithm can achieve certain accuracy with lower computational complexity. Finally, numerical experiments were conducted to demonstrate the theoretical results and the advantages of the algorithm on computation cost. In future works, we wish to further explore and account for the heterogeneity of the hyerpergradients among nodes.

VI. Appendix

A. Proof of Lemma 1

According to the Lipschitz continuity of $\nabla\Phi$ and the definition of the estimate $\bar{s}^k$, it follows that:

$$\begin{aligned}
&\mathbb{E}\left[\Phi(\bar{x}^{k+1})|\mathcal{F}^k\right]\\
\leq&\Phi(\bar{x}^k) - \alpha\mathbb{E}\left[\left\langle\nabla\Phi(\bar{x}^k), \bar{s}^k\right\rangle|\mathcal{F}^k\right] + \frac{\alpha^2 L}{2}\mathbb{E}[\|\bar{s}^k\|^2|\mathcal{F}^k]\\
=&\Phi(\bar{x}^k) - \frac{\alpha}{2}\|\nabla\Phi(\bar{x}^k)\|^2 - \frac{\alpha}{2}\left(1-\alpha L\right)\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2\\
&+\frac{\alpha}{2}\left\|\nabla\Phi(\bar{x}^k) - \mathbb{E}[\bar{s}^k|\mathcal{F}^k]\right\|^2 + \frac{\alpha^2 L}{2}\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k] - \bar{s}^k\|^2|\mathcal{F}^k],
\end{aligned} \tag{19}$$

where we use the recursion (6a) in the first inequality. ■

B. Proof of Lemma 2

Recall the recursion of s_i^k by (6c) and note that

$$\mathbb{E}[s_i^k|\mathcal{F}^k] = \nabla_x f_i\left(x_i^k, \theta_i^k\right) - \nabla_{x\theta}^2 g_i\left(x_i^k, \theta_i^k\right) v_i^k. \tag{20}$$

Then, using Assumption 3 and the fact that $\|v_i^k\| \leq M$ in Proposition 4, we have

$$\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k] - \bar{s}^k\|^2|\mathcal{F}^k] \leq \frac{1}{m}(\sigma^2 + M^2\sigma^2). \tag{21}$$

Then, combining the definition of the hypergradient $\nabla\Phi(\bar{x}^k)$ and the equality (4) and using the fact that $v_{\theta_i^*(\bar{x}^k)} = \left[\nabla_{\theta\theta}^2 g_i\left(\bar{x}^k, \theta_i^*(\bar{x}^k)\right)\right]^{-1}\nabla_\theta f\left(\bar{x}^k, \theta_i^*(\bar{x}^k)\right)$, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla\Phi(\bar{x}^k) - \mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2|\mathcal{F}^k]\\
\leq&2\frac{1}{m}\sum_{i=1}^m\|\nabla_x f_i(\bar{x}^k, \theta_i^*(\bar{x}^k)) - \nabla_x f_i\left(x_i^k, \theta_i^k\right)\|^2\\
&+4\frac{1}{m}\sum_{i=1}^m\|\nabla_{x\theta}^2 g_i\left(\bar{x}^k, \theta_i^*(\bar{x}^k)\right)\left(v_i^k - v_{\theta_i^*(\bar{x}^k)}\right)\|^2\\
&+4\frac{1}{m}\sum_{i=1}^m\|\left(\nabla_{x\theta}^2 g_i\left(x_i^k, \theta_i^k\right) - \nabla_{x\theta}^2 g_i\left(\bar{x}^k, \theta_i^*(\bar{x}^k)\right)\right)v_i^k\|^2\\
\leq&D\frac{1}{m}\sum_{i=1}^m\|x_i^k - \bar{x}^k\|^2 + D\frac{1}{m}\sum_{i=1}^m\|\theta_i^k - \theta_i^*(\bar{x}^k)\|^2\\
&+4C_{g,x\theta}^2\frac{1}{m}\sum_{i=1}^m\|v_i^k - v_{\theta_i^*(\bar{x}^k)}\|^2,
\end{aligned} \tag{22}$$

where $D = 2L_{f,x}^2 + 4M^2L_{g,x\theta}^2$, and the last inequality follows from Assumption 1(ii), Lemmas 1 and 4 and Assumption 2(ii). Finally, taking the total expectation of both sides of the above inequality completes the proof. ■

C. Proof of Lemma 3

Note that the term $\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_i^{k+1} - v_{\theta_i^*(\bar{x}^{k+1})}\|^2|\mathcal{F}^k]$ can be expanded as:

$$\begin{aligned}
&\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_i^{k+1} - v_{\theta_i^*(\bar{x}^{k+1})}|\mathcal{F}^k\|^2]\\
=&\frac{1}{m}\sum_{i=1}^m\underbrace{\mathbb{E}[\|v_i^{k+1} - v_{\theta_i^*(\bar{x}^k)}\|^2|\mathcal{F}^k]}_{\triangleq V}\\
&+\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|v_{\theta_i^*(\bar{x}^k)} - v_{\theta_i^*(\bar{x}^{k+1})}\|^2|\mathcal{F}^k]\\
&+\frac{1}{m}\sum_{i=1}^m\underbrace{2\mathbb{E}[\langle v_i^{k+1} - v_{\theta_i^*(\bar{x}^k)}, v_{\theta_i^*(\bar{x}^k)} - v_{\theta_i^*(\bar{x}^{k+1})}\rangle|\mathcal{F}^k]}_{\triangleq Z}.
\end{aligned} \tag{23}$$

Using the update (6c) and $v_{\theta_i^*(\bar{x}^k)} = \left[\nabla_{\theta\theta}^2 g_i\left(\bar{x}^k, \theta_i^*(\bar{x}^k)\right)\right]^{-1}\nabla_x f_i(\bar{x}^k, \theta_i^*(\bar{x}^k))$ and introducing the term $\left(I - \lambda\nabla_{\theta\theta}^2 g_i(x_i^k, \theta_i^k)\right)v_i^k + \lambda\nabla_\theta f_i\left(x_i^k, \theta_i^k\right)$, the first term can be split as

$$\begin{aligned}
V =&\underbrace{\mathbb{E}\left[\left\|\begin{array}{l}\left(I - \lambda\nabla_{\theta\theta}^2\hat{g}_i(x_i^k, \theta_i^k; \xi_{i,2}^{k+1})\right)v_i^k\\ -\left(I - \lambda\nabla_{\theta\theta}^2 g_i(x_i^k, \theta_i^k)\right)v_i^k\\ +\lambda\left(\nabla_\theta\hat{f}_i(x_i^k, \theta_i^k; \varsigma_{i,1}^{k+1}) - \nabla_\theta f_i(x_i^k, \theta_i^k)\right)\end{array}\right\|^2|\mathcal{F}^k\right]}_{\triangleq V_1}\\
&+\underbrace{\left\|\begin{array}{c}\left(I - \lambda\nabla_{\theta\theta}^2 g_i(x_i^k, \theta_i^k)\right)v_i^k\\ -\left(I - \lambda\nabla_{\theta\theta}^2 g_i(\bar{x}^k, \theta_i^*(\bar{x}^k))\right)v_{\theta_i^*(\bar{x}^k)}\\ +\lambda\left(\nabla_\theta f_i\left(x_i^k, \theta_i^k\right) - \nabla_\theta f_i(\bar{x}^k, \theta_i^*(\bar{x}^k))\right)\end{array}\right\|^2}_{\triangleq V_2}.
\end{aligned} \tag{24}$$

The variance term V_1 in (24) can be bounded as

$$V_1 \leq \left(1+M^2\right)\lambda^2\sigma^2. \tag{25}$$

For the term V_2 in (24), it follows from the Lipsthiz continuity of $\nabla^2_{\theta\theta}g_i$ and $\nabla_\theta f_i$ as well as Proposition 4 that

$$\begin{aligned}
V_2 \leq & \left(1+\frac{\mu_g\lambda}{2}\right)\left\|\begin{array}{l}\left(I-\lambda\nabla^2_{\theta\theta}g_i(x_i^k,\theta_i^k)\right)\left(v_i^k-v_{\theta_i^*(\bar{x}^k)}\right)\\ +\left(I-\lambda\nabla^2_{\theta\theta}g_i(x_i^k,\theta_i^k)\right)v_{\theta_i^*(\bar{x}^k)}\\ -\left(I-\lambda\nabla^2_{\theta\theta}g_i(\bar{x}^k,\theta_i^*(\bar{x}^k))\right)v_{\theta_i^*(\bar{x}^k)}\end{array}\right\|^2\\
& +\left(1+\frac{2}{\mu_g\lambda}\right)\left\|\begin{array}{l}\lambda\nabla_\theta f_i\left(x_i^k,\theta_i^k\right)\\ -\lambda\nabla_\theta f_i\left(\bar{x}^k,\theta_i^*(\bar{x}^k)\right)\end{array}\right\|^2\\
\leq & (1-\lambda\mu_g)\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|^2\\
& \frac{3B\lambda}{\mu_g}[\|x_i^k-\bar{x}^k\|^2+\|\theta_i^k-\theta_i^*(\bar{x}^k)\|^2],
\end{aligned} \tag{26}$$

where $B=2L^2_{g,\theta\theta}M^2+L^2_{f,\theta}$. Next we proceed to bound the term Z in (23). Note that by the recursion of v_i^{k+1} in (6c), we can expand the term Z as:

$$\begin{aligned}
Z = & \underbrace{2\mathbb{E}[\langle v_i^k-v_{\theta_i^*(\bar{x}^k)},v_{\theta_i^*(\bar{x}^k)}-v_{\theta_i^*(\bar{x}^{k+1})}\rangle}_{\triangleq Z_1}|\mathcal{F}^k]\\
& +\underbrace{2\lambda\mathbb{E}[\langle\mathbb{E}[v_{\theta_i^k}|\mathcal{F}^k],v_{\theta_i^*(\bar{x}^k)}-v_{\theta_i^*(\bar{x}^{k+1})}\rangle|\mathcal{F}^k]}_{\triangleq Z_2},
\end{aligned} \tag{27}$$

where $v_{\theta_i^k}\triangleq\nabla_\theta\hat{f}_i(x_i^k,\theta_i^k;\varsigma_{i,1}^{k+1})-\nabla^2_{\theta\theta}\hat{g}_i(x_i^k,\theta_i^k;\xi_{i,2}^{k+1})v_i^k$. Then we can bound the term Z_1 as:

$$\begin{aligned}
& Z_1\\
=&2\mathbb{E}[\langle v_i^k-v_{\theta_i^*(\bar{x}^k)},v_{\theta_i^*(\bar{x}^k)}-v_{\theta_i^*(\bar{x}^{k+1})}-\nabla v_{\theta_i^*(\bar{x}^k)}(\bar{x}^k-\bar{x}^{k+1})\rangle|\mathcal{F}^k]\\
& +2\mathbb{E}[\langle v_i^k-v_{\theta_i^*(\bar{x}^k)},\nabla v_{\theta_i^*(\bar{x}^k)}(\bar{x}^k-\bar{x}^{k+1})|\mathcal{F}^k]\rangle\\
\leq&\alpha^2L_{v,x}\mathbb{E}[\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|\|\bar{s}^k\|^2|\mathcal{F}^k]\\
& +2\alpha\|\nabla v_{\theta_i^*(\bar{x}^k)}\|\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|\\
\leq&\alpha^2\mathbb{E}[\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|^2\|\bar{s}^k\|^2|\mathcal{F}^k]+L^2_{v,x}\alpha^2\mathbb{E}[\|\bar{s}^k\|^2|\mathcal{F}^k]\\
& +\lambda\varpi\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|^2+\frac{L_v^2\alpha^2}{\lambda\varpi}\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2\\
\leq&(U^2\alpha^2+\lambda\varpi)\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|^2+(\frac{L^2_{v,x}}{L_v^2}\lambda\varpi+1)\frac{L_v^2\alpha^2}{\lambda\varpi}\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2\\
& +(1+M^2)L^2_{v,x}\alpha^2\sigma^2,
\end{aligned} \tag{28}$$

where *i)* the first inequality follows from the Lipschitz continuity of $\nabla v_{\theta_i^*(\bar{x}^k)}$ and Cauchy-Schwarz inequality; *ii)* the second inequality uses the basic inequality $2ab\leq ta^2+\frac{1}{t}b^2$ for any $t>0$ and the unbiased estimate of $\bar{s}^k$, and we take $\varpi=\frac{\mu_g}{2}$; *iii)* the last inequality comes from (21) and the following bound for $\mathbb{E}[\|\bar{s}^k\|^2|\mathcal{F}^k]$:

$$\mathbb{E}[\|\bar{s}^k\|^2|\mathcal{F}^k]\leq U\triangleq 2\left(C^2_{f,x}+C^2_{g,x\theta}M\right)+\left(1+M^2\right)\sigma^2. \tag{29}$$

Now we deal with the term Z_2 in (27). Note that by utilizing $v_{\theta_i^*(\bar{x}^k)}$ and Assumption 3, the term $\mathbb{E}[v_{\theta_i^k}|\mathcal{F}^k]$ of Z_2 can be split into:

$$\begin{aligned}
\mathbb{E}[v_{\theta_i^k}|\mathcal{F}^k]=&-\nabla^2_{\theta\theta}g_i(x_i^k,\theta_i^k)(v_i^k-v_{\theta_i^*(\bar{x}^k)})\\
& -(\nabla^2_{\theta\theta}g_i(x_i^k,\theta_i^k)-\nabla^2_{\theta\theta}g_i(\bar{x}^k,\theta_i^*(\bar{x}^k)))v_{\theta_i^*(\bar{x}^k)}\\
& +\nabla_\theta f_i(x_i^k,\theta_i^k)-\nabla_\theta f_i(\bar{x}^k,\theta_i^*(\bar{x}^k)).
\end{aligned} \tag{30}$$

Then, applying the basic inequality $2ab\leq ta^2+\frac{1}{t}b^2$ for any $t>0$ and using Assumptions 1-3 and Proposition 4, the term Z_2 can be bounded by:

$$\begin{aligned}
& Z_2\\
\leq&\lambda^2L^2_{g,\theta}\|v_i^k-v_{\theta_i^*(\bar{x}^k)}\|^2+3L_v^2\alpha^2\left\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\right\|^2\\
& +\lambda^2\left(M^2L^2_{g,\theta\theta}+L^2_{f,\theta}\right)[\|x_i^k-\bar{x}^k\|^2+\|\theta_i^k-\theta_i^*(\bar{x}^k)\|^2]\\
& +3(1+M^2)L_v^2\alpha^2\sigma^2.
\end{aligned} \tag{31}$$

Combining the fact that $\mathbb{E}[\|v_{\theta_i^*(\bar{x}^k)}-v_{\theta_i^*(\bar{x}^{k+1})}\|^2|\mathcal{F}^k]\leq L_v^2\alpha^2\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2+(1+M^2)L_v^2\alpha^2\sigma^2$ and substituting (25), (26), (28), (31) into (23), we can derive the derived result (13), which completes the proof. ■

D. Proof of Lemma 4

Akin to the inequality (23), we can expand the term $\frac{1}{m}\sum_{i=1}^m\mathbb{E}[\|\theta_i^{k+1}-\theta_i^*(\bar{x}^{k+1})\|^2|\mathcal{F}^k]$ according to following three terms: $\mathbb{E}[2\langle\theta_i^{k+1}-\theta_i^*(\bar{x}^k),\theta_i^*(\bar{x}^k)-\theta_i^*(\bar{x}^{k+1})\rangle|\mathcal{F}^k]$, $\mathbb{E}[\|\theta_i^{k+1}-\theta_i^*(\bar{x}^k)\|^2|\mathcal{F}^k]$ and $\mathbb{E}[\|\theta_i^*(\bar{x}^k)-\theta_i^*(\bar{x}^{k+1})\|^2|\mathcal{F}^k]$. Firstly, we aim to bound the term $\mathbb{E}[\|\theta_i^{k+1}-\theta_i^*(\bar{x}^k)\|^2|\mathcal{F}^k]$. To this end, we start by expressing the term $\theta_i^{k+1}-\theta_i^*(\bar{x}^k)$ as:

$$\begin{aligned}
\theta_i^{k+1}-\theta_i^*(\bar{x}^k)=&\underbrace{\theta_i^k-\beta\nabla_\theta g_i(\bar{x}^k,\theta_i^k)-\theta_i^*(\bar{x}^k)}_{\triangleq Q_1}\\
& +\beta\underbrace{\left(\nabla_\theta g_i(\bar{x}^k,\theta_i^k)-\nabla_\theta\hat{g}_i(x_i^k,\theta_i^k;\xi_{i,1}^{k+1})\right)}_{\triangleq Q_2}.
\end{aligned} \tag{32}$$

We then provide the upper bounds for the last two terms of (32). Specifically, for the first term Q_1, we have

$$\begin{aligned}
\|Q_1\|^2=&\left\|\theta_i^k-\theta_i^*(\bar{x}^k)\right\|^2+\left\|\nabla_\theta g_i(\bar{x}^k,\theta_i^k)\right\|^2\\
& -2\beta\left\langle\theta_i^k-\theta_i^*(\bar{x}^k),\nabla_\theta g_i(\bar{x}^k,\theta_i^k)\right\rangle\\
\leq&c_1\left\|\theta_i^k-\theta_i^*(\bar{x}^k)\right\|^2+c_2\left\|\nabla_\theta g_i(\bar{x}^k,\theta_i^k)\right\|^2\\
\leq&c_1\left\|\theta_i^k-\theta_i^*(\bar{x}^k)\right\|^2,
\end{aligned} \tag{33}$$

where $c_1\triangleq 1-2\beta\frac{\mu_gL_{g,\theta}}{\mu_g+L_{g,\theta}}$, $c_2\triangleq\beta^2-2\beta\frac{1}{\mu_g+L_{g,\theta}}$, and the first inequality is derived according to the strong convexity and Lipschitz-continuous gradient of g_i and the last inequality holds due to $\beta<\frac{2}{\mu_g+L_{g,\theta}}$. In conjunction with (32) and (33), the term $\mathbb{E}[\|\theta_i^{k+1}-\theta_i^*(\bar{x}^k)\|^2|\mathcal{F}^k]$ can be bounded by:

$$\begin{aligned}
& \mathbb{E}[\|\theta_i^{k+1}-\theta_i^*(\bar{x}^k)\|^2|\mathcal{F}^k]\\
=&\|Q_1\|^2+\mathbb{E}[\|\beta Q_2\|^2|\mathcal{F}^k]+2\beta\langle\mathbb{E}[Q_2|\mathcal{F}^k],Q_1\rangle\\
\leq&(1-\beta\frac{\mu_gL_{g,\theta}}{\mu_g+L_{g,\theta}})\|\theta_i^k-\theta_i^*(\bar{x}^k)\|^2\\
& +(\beta+\frac{\mu_g+L_{g,\theta}}{\mu_gL_{g,\theta}})\beta L^2_{g,\theta}\|x_i^k-\bar{x}^k\|^2+\beta^2\sigma^2,
\end{aligned} \tag{34}$$

where the last inequality is derived according to Cauchy-Schwarz inequality and Lipshitz continuity of $\nabla_\theta g_i$.
For the term $\mathbb{E}[2\langle\theta_i^{k+1}-\theta_i^*(\bar{x}^k),\theta_i^*(\bar{x}^k)-\theta_i^*(\bar{x}^{k+1})\rangle|\mathcal{F}^k]$, an upper bound can be established in a similar way as obtaining inequalities (28) and (31) as follows:

$$\begin{aligned}&\mathbb{E}[2\langle\theta_i^{k+1}-\theta_i^*(\bar{x}^k),\theta_i^*(\bar{x}^k)-\theta_i^*(\bar{x}^{k+1})\rangle|\mathcal{F}^k]\\ \leq&\big(U^2\alpha^2+\omega\beta+\beta^2L_{g,\theta}^2\big)\|\theta_i^k-\theta_i^*(\bar{x}^k)\|^2+\beta^2L_{g,\theta}^2\|x_i^k-\bar{x}^k\|^2\\ &+((1+\frac{L_{\theta^*,x}^2}{L_{\theta^*}^2})\omega\beta+1)\frac{L_{\theta^*}^2\alpha^2}{\beta\omega}\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2\\ &+(1+M^2)(L_{\theta^*}^2+L_{\theta^*,x}^2)\alpha^2\sigma^2,\end{aligned}\tag{35}$$

where $\omega\triangleq\frac{\mu_gL_{g,\theta}}{2(\mu_g+L_{g,\theta})}$. Then, combining the fact that $\mathbb{E}[\|\theta_i^*(\bar{x}^k)-\theta_i^*(\bar{x}^{k+1})\|^2|\mathcal{F}^k]\leq L_{\theta^*}^2\alpha^2\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2+(1+M^2)L_{\theta^*}^2\alpha^2\sigma^2$, and using the inequalities (34) and (35), we obtain the desired result (14), which completes the proof. ■

E. Proof of Lemma 5

For notation simplicity, we let $x^k\in\mathbb{R}^{mn}$ and $s^k\in\mathbb{R}^{mn}$ respectively denote the vector stackings the corresponding local vectors x_i^k and local gradients s_i^k. Thus, for all $i\in\mathcal{V}$, the recursion (6a) can be compactly rewritten as

$$x^{k+1}=(W\otimes I_n)x^k-\alpha s^k.$$

By combining the update of the inner variables and using the relaxed triangle inequality with parameters $\eta=\frac{1-\rho}{2\rho}$ and $\rho=\|(W-J)\otimes I_n\|^2\in[0,1)$, we have

$$\begin{aligned}&\mathbb{E}[\|x^{k+1}-1_m\otimes\bar{x}^{k+1}\|^2|\mathcal{F}^k]\\ \leq&\frac{1+\rho}{2}\|x^k-1_m\otimes\bar{x}^k\|^2+\frac{2mU\alpha^2}{1-\rho},\end{aligned}\tag{36}$$

where we use the inequality (29). In what follows, utilizing the recursive expression of the above inequality with an initialization $\left\|x^0-1_m\otimes\bar{x}^0\right\|^2=0$ and summing the resulting series of inequalities, we can obtain the desired result (15) from (36), which completes the proof. ■

F. Proof of Theorem 1

Utilizing the definition of the potential function Δ^k, integrating the inequalities (10), (11), (13), (14), (15) and rearranging the terms, we can reach

$$\begin{aligned}&\mathbb{E}[\|\nabla\Phi(\bar{x}^k)\|^2]\\ \leq&\frac{2\big(\Delta^k-\Delta^{k+1}\big)}{c_\lambda\mu_g\Xi\alpha}+r_1\alpha\sigma^2+\frac{r_2\alpha^2}{(1-\rho)^2}-\delta\mathbb{E}[\|\mathbb{E}[\bar{s}^k|\mathcal{F}^k]\|^2],\end{aligned}$$

where $\delta\triangleq1-\alpha L-\frac{2\left(8C_{g,x\theta}^2\Xi q_s+\Gamma p_s\right)}{c_\lambda\mu_g\Xi}\alpha$, $r_1\triangleq\frac{2}{c_\lambda\mu_g\Xi\alpha}\left(\left(\frac{c_\lambda\mu_gL}{2m}\left(1+M^2\right)+8C_{g,x\theta}^2q_\sigma\right)\Xi+p_\sigma\Gamma\right)$, $r_2\triangleq\frac{8U}{c_\lambda\mu_g\Xi\alpha}\left(\left(\frac{c_\lambda\mu_gD}{2}+8C_{g,x\theta}^2q_x\right)\Xi+p_x\Gamma\right)$. Then, when the step-size satisfies $\alpha\leq u$ with u defined in (17) such that $\delta>0$, summing the above inequality over $k=0,\cdots,K$ yields

$$\frac{1}{K+1}\sum_{k=0}^{K}\mathbb{E}[\|\nabla\Phi(\bar{x}^k)\|^2]\leq\frac{r_0}{(K+1)\alpha}+r_1\alpha\sigma^2+r_2\frac{\alpha^2}{(1-\rho)^2}.$$

with $r_0=\frac{2(\Delta_0-\Delta_K)}{c_\lambda\mu_g\Xi}$. If the step-size α is further set as $\alpha=\min\left(u,\left(\frac{r_0}{r_1\sigma^2(K+1)}\right)^{\frac{1}{2}},\left(\frac{r_0(1-\rho)^2}{r_2(K+1)}\right)^{\frac{1}{3}}\right)$, we can obtain the result (18) by employing the similar steps from Theorem 1 in [11], which completes the proof. ■

REFERENCES

[1] N. Zucchet and J. Sacramento, "Beyond backpropagation: implicit gradients for bilevel optimization," *arXiv preprint arXiv:2205.03076*, 2022.

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

[3] M. Hong, H.-T. Wai, Z. Wang, and Z. Yang, "A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic," *SIAM Journal on Optimization*, vol. 33, no. 1, pp. 147–180, 2023.

[4] S. Lu, X. Cui, M. S. Squillante, B. Kingsbury, and L. Horesh, "Decentralized bilevel optimization for personalized client learning," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5543–5547, IEEE, 2022.

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

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

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

[8] J. Li, B. Gu, and H. Huang, "A fully single loop algorithm for bilevel optimization without hessian inverse," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 7426–7434, 2022.

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

[10] M. Dagréou, P. Ablin, S. Vaiter, and T. Moreau, "A framework for bilevel optimization that enables stochastic and global variance reduction algorithms," *arXiv preprint arXiv:2201.13409*, 2022.

[11] Y. Chen, K. Yuan, Y. Zhang, P. Pan, Y. Xu, and W. Yin, "Accelerating gossip SGD with periodic global averaging," in *International Conference on Machine Learning*, pp. 1791–1802, PMLR, 2021.

[12] H. Li, Q. Lü, and T. Huang, "Distributed projection subgradient algorithm over time-varying general unbalanced directed graphs," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1309–1316, 2018.

[13] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the admm in decentralized consensus optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1750–1761, 2014.

[14] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes," in *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 2055–2060, IEEE, 2015.

[15] Y. Sun, G. Scutari, and A. Daneshmand, "Distributed optimization based on gradient tracking revisited: Enhancing convergence rate via surrogation," *SIAM Journal on Optimization*, vol. 32, no. 2, pp. 354–385, 2022.

[16] X. Chen, M. Huang, and S. Ma, "Decentralized bilevel optimization," *arXiv preprint arXiv:2206.05670*, 2022.

[17] S. Yang, X. Zhang, and M. Wang, "Decentralized gossip-based stochastic bilevel optimization over communication networks," *arXiv preprint arXiv:2206.10870*, 2022.

[18] H. Gao, B. Gu, and M. T. Thai, "Stochastic bilevel distributed optimization over a network," *arXiv preprint arXiv:2206.15025*, 2022.

[19] K. Ji, M. Liu, Y. Liang, and L. Ying, "Will bilevel optimizers benefit from loops," *arXiv preprint arXiv:2205.14224*, 2022.

[20] X. Chen, M. Huang, S. Ma, and K. Balasubramanian, "Decentralized stochastic bilevel optimization with improved per-iteration complexity," *arXiv preprint arXiv:2210.12839*, 2022.

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