UNIVERSITY OF CALIFORNIA

Los Angeles

Stochastic Optimization Methods for Modern Machine Learning Problems

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Mathematics

by

Yuejiao Sun

2021

ABSTRACT OF THE DISSERTATION

Stochastic Optimization Methods for Modern Machine Learning Problems

by

Yuejiao Sun
Doctor of Philosophy in Mathematics
University of California, Los Angeles, 2021
Professor Wotao Yin, Chair

Optimization has been the workhorse of solving machine learning problems. The present dissertation will focus on two fundamental classes of machine learning problems: 1) stochastic nested problems, where one subproblem builds upon the solution of others; and, 2) stochastic distributed problems, where the subproblems are coupled through sharing the variables.

One key difficulty of solving stochastic nested problems is that the hierarchically coupled structure makes the computation of (stochastic) gradients, the basic element in first-order optimization machinery, prohibitively expensive or even impossible. We will develop the *first* stochastic optimization method, which runs in a single-loop manner and achieves the same sample complexity as the stochastic gradient descent method for non-nested problems.

One key difficulty of solving stochastic distributed problems is the resource intensity, especially when algorithms are running at resource-limited devices. In this context, we will introduce a class of communication-adaptive stochastic gradient descent (SGD) methods, which adaptively reuse the stale gradients, thus saving communication. We will show that the new algorithms have convergence rates comparable to original SGD and Adam algorithms, but enjoy impressive empirical performance in terms of total communication round reduction.

The dissertation of Yuejiao Sun is approved.

Guido Montúfar

Deanna Needell

Qing Zhou

Wotao Yin, Committee Chair

University of California, Los Angeles

2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First I would like to express my deepest gratitude to my advisor, Prof. Wotao Yin. His invaluable guidance on research directions, expertise on methology and feedback on presentations has been extraordinary to me and made me an independent researcher. His constant encouragement and patience has always been a great support during the time of my academic research.

I would also like to express my special thanks to my committee members, Prof. Guido Montúfar, Prof. Deanna Needell and Prof. Qing Zhou. Their insightful comments, valuable criticism, and constant encouragement have greatly improved the quality of my work.

Also it was a great fortune for me to have opportunities to collaborate with excellent people during my PhD study. First I would like to thank Prof. Tianyi Chen. His insightful suggestions and continuous patience during the collaboration have affected me a lot and made be a better researcher. And I would like to express my gratitude to my mentors, Dr. Brendt Wohlberg and Dr. Cristina Garcia-Cardona, when I interned at Los Alamos National Laboratory. Their broad knowledge has made this an inpiring experience for me. I would also like to extend due credit and warmest thanks to Dr. Tao Sun, Dr. Yanli Liu, Yifan Chen, Prof. Yangyang Xu, Ziye Guo and Xiao Jin for their insightful input to our collaborations.

This thesis has also benefited from discussions with my colleges: Dr. Hanqing Cai, Fei Feng, Dr. Samy Wu Fung, Dr. Robert Hannah, Howard Heaton, Bumsu Kim, Dr. Qiuwei Li, Dr. Jialin Liu, Siting Liu, Dr. Daniel McKenzie, Prof. Ernest Ryu, Dr. Kun Yuan. This thesis would not be possible without their helpful participation.

Finally, I would like to thank my parents for their unconditional love and support along the way. Thank you for always being there for me, with a sympathetic ear and gentle encouragement.

VITA

2012–2016	Bachelor of Science in Mathematics Peking University, Beijing, China
2016–2021	Teaching and Research Assistant, Department of Mathematics University of California - Los Angeles, Los Angeles, CA, USA
2018	Research Intern Los Alamos National Laboratory, Santa Fe, NM, USA
2020	Research Intern Alibaba DAMO Academy, Seattle, WA, USA

PUBLICATIONS

Tianyi Chen, Ziye Guo, Yuejiao Sun and Wotao Yin, CADA: Communication-Adaptive Distributed Adam. *International Conference on Artificial Intelligence and Statistics* (AISTATS), 2021.

Tao Sun, Yuejiao Sun, Yangyang Xu and Wotao Yin. Markov Chain Block Coordinate Descent. *Computational Optimization and Applications*, 2020: 1-27.

Yifan Chen, Yuejiao Sun and Wotao Yin, Run-and-Inspect Method for Nonconvex Optimization and Global Optimality Bounds for R-local Minimizers. *Mathematical Programming*, 2019: 176(1-2), 39-67.

Tao Sun, Yuejiao Sun and Wotao Yin, On Markov Chain Gradient Descent. *Advances in Neural Information Processing Systems* (NeurIPS), 2018: 9896-9905.

Yanli Liu, Yuejiao Sun and Wotao Yin. Decentralized Learning with Lazy and Approximate Dual Gradients. *IEEE Transactions on Signal Processing*, vol. 69, 2021.

Tianyi Chen, Yuejiao Sun, Wotao Yin. A Single-Timescale Stochastic Bilevel Optimization Method. *arXiv preprint arXiv:2102.04671*, 2021.

Tianyi Chen, Yuejiao Sun, Wotao Yin. Solving Stochastic Compositional Optimization is Nearly as Easy as Solving Stochastic Optimization. *arXiv preprint arXiv:2008.10847*, 2020.

Tianyi Chen, Yuejiao Sun, Wotao Yin. LASG: Lazily Aggregated Stochastic Gradients For Communication-Efficient Distributed Learning. *arXiv preprint arXiv:2002.11360*, 2020.

CHAPTER 1

Introduction

Optimization has been the workhorse of solving machine learning problems. However, the efficiency of these methods remains far from satisfaction to meet the ever-growing demand that arises in modern machine learning applications. In these applications, the models are nonconvex and nonsmooth, tasks are coupled or nested, the data are distributed among agents, which make the theoretically optimal methods often practically inefficient, or even lose their desired analytical guarantees. A timely opportunity thus emerges to transform the ordinary optimization framework into a contemporary one tailored for modern machine learning. My dissertation contributes to this transformative research area.

In this context, the present dissertation will mainly focus on developing new stochastic optimization algorithms to tackle two fundamental structures of optimization problems: C1) stochastic nested optimization problems, where one subproblem builds upon the solution of others; and C2) stochastic distributed optimization problems, where the subproblems are coupled through sharing the common variables. These two problem structures capture various machine learning problems.

1.1 Stochastic nested optimization

In the first part of the thesis, which contains Chapters 2 and 3, the goal is to develop sample-efficient stochastic optimization methods amenable to solve stochastic nested problems in C1. This part is based on the publications [11, 12]. The stochastic nested problems considered in this thesis can be summarized as

$$\begin{aligned} \min_{\theta \in \mathbb{R}^d} \quad & F(\theta) := \mathbb{E}_\xi[f(\theta, y^*(\theta); \xi] \\ \text{s.t.} \quad & y^*(\theta) = \arg\min_{y \in \mathbb{R}^{d'}} \mathbb{E}_\phi[g(\theta, y; \phi)] \end{aligned} \tag{1.1}$$

where f and g are differentiable functions; and, ξ and ϕ are random variables. The above problem is often referred to as the stochastic *bilevel* problem, where the upper-level optimization problem depends on the solution of the lower-level optimization over $y \in \mathbb{R}^{d'}$, denoted as $y^*(\theta)$, which depends on the value of upper-level variable $\theta \in \mathbb{R}^d$.

Stochastic bilevel optimization generalizes the classic stochastic optimization from the minimization of a single objective to the minimization of an objective function that depends the solution of another optimization problem. Examples of stochastic nested problems include model-agnostic meta learning (MAML), where the goal is to find a model that not only achieves a good average performance on the training set, but also can quickly adapt to a new data set in the testing stage [26]. Likewise, stochastic nested problems also emerge in reinforcement learning (RL), where finding the optimal policy requires to estimate the quality of a given policy, but such a quality estimate can only be obtained by solving a policy evaluation subproblem [56]. One key difficulty of solving this class of problems is that the nested structure makes the computation of (stochastic) gradients, the basic element in first-order optimization machinery, prohibitively expensive or even impossible.

To get some insights, we will first tackle a special case of (1.1). When the lower-level problem is $y^*(\theta) = \arg\min_{y \in \mathbb{R}^{d'}} \mathbb{E}_\phi[\|y - g(\theta; \phi)\|^2]$, the bilevel problem (1.1) reduces to the stochastic single-level yet compositional optimization, given by

$$\min_{\theta \in \mathbb{R}^d} \quad F(\theta) := \mathbb{E}_\xi[f(\theta, \mathbb{E}_\phi[g(\theta; \phi)]; \xi]. \tag{1.2}$$

Stochastic compositional optimization generalizes classic (non-compositional) stochastic optimization to the minimization of compositions of functions. Each composition may introduce an additional expectation. Chpater 2 presents a Stochastically Corrected Stochastic

Compositional gradient method (**SCSC**). SCSC runs in a single-time scale with a single loop, uses a fixed batch size, and guarantees to converge to an ϵ-stationary point using $\mathcal{O}(\epsilon^{-2})$ samples in total, which is the same as the stochastic gradient descent (SGD) method for non-compositional stochastic optimization. It is easy to apply SGD-improvement techniques to accelerate SCSC. This helps SCSC achieve state-of-the-art performance for stochastic compositional optimization. In particular, we apply Adam to SCSC, and the exhibited rate of convergence matches that of the original Adam on non-compositional stochastic optimization.

To tackle the general stochastic bilevel problem (1.1), existing methods require either double-loop or two-timescale updates, which are sometimes less efficient. In Chapter 3, we develop a new optimization method for a class of stochastic bilevel problems that we term Single-Timescale stochAstic BiLevEl optimization (**STABLE**) method. STABLE runs in a single loop fashion, and uses a single-timescale update with a fixed batch size. To achieve an ϵ-stationary point of the bilevel problem, STABLE requires $\mathcal{O}(\epsilon^{-2})$ samples in total; and to achieve an ϵ-optimal solution in the strongly convex case, STABLE requires $\mathcal{O}(\epsilon^{-1})$ samples. To the best of our knowledge, this is the *first* bilevel optimization algorithm achieving the same order of sample complexity as the stochastic gradient descent method for the single-level stochastic optimization.

1.2 Stochastic distributed optimization

In the second part of the thesis, which contains Chapters 4 and 5, the aim was to develop communication-efficient distributed stochastic optimization methods amenable to solve stochastic distributed problems in C2. This part is base on the publications [10, 9]. The key take-home message there is that by exploiting the informativeness of message, our new distributed stochastic optimization methods can achieve the same convergence rate but save significantly communication overhead.

The stochastic distributed problems considered in this thesis can be summarized as

$$\min_{\theta\in\mathbb{R}^d} \quad \mathcal{L}(\theta)=\frac{1}{M}\sum_{m\in\mathcal{M}}\mathcal{L}_m(\theta) \quad \text{with} \quad \mathcal{L}_m(\theta):=\mathbb{E}_{\xi_m}\left[\ell(\theta;\xi_m)\right],\; m\in\mathcal{M} \tag{1.3}$$

where $\mathcal{M}=\{1,2\ldots,M\}$ is the collection of multiple computing nodes, and ξ_m is the (random) local data on node m. Examples of stochastic distributed problems in C2 include distributed learning or recently federated learning [72]. In this case, the data as well as the parameters are kept on local nodes (e.g., mobile devices), and only the changes of model parameters will be communicated between computing nodes. There one of the main challenges is the resource intensity, because machine learning tasks are running at wirelessly connected devices which are often resource-constrained. Very often, resource scarcity, amplified by data and system heterogeneity, will become the obstacle of further improving machine learning performance.

Chapter 4 develops algorithms for solving distributed learning problems in a communication-efficient fashion, by generalizing the recent method of lazily aggregated gradient (LAG) to deal with stochastic gradient — justifying the name of the new method LASG. While LAG is effective at reducing communication without sacrificing the rate of convergence, we show it only works with deterministic gradients. We introduce new rules and analysis for LASG that are tailored for SGD, so it effectively saves downloads, uploads, or both for distributed SGD. LASG achieves impressive empirical performance — it typically saves total communication by an order of magnitude.

In practice, SGD is often used with its adaptive variants such as AdaGrad, Adam, and AMSGrad. Chapter 5 proposes an adaptive SGD method for distributed machine learning, which can be viewed as the communication-adaptive counterpart of the celebrated Adam method — justifying its name CADA. The key components of CADA are a set of new rules tailored for adaptive SGD that can be implemented to save communication upload. The new algorithms adaptively reuse the stale Adam gradients, thus saving communication, and still have convergence rates comparable to original Adam. In numerical experiments, CADA achieves impressive empirical performance in terms of total communication round reduction.

CHAPTER 2

Single-loop Algorithms for Stochastic Compositional Optimization

2.1 Introduction

In this chapter, we consider stochastic compositional optimization problems of the form

$$\min_{\theta\in\mathbb{R}^d} \quad F(\theta) := f_N\left(f_{N-1}(\cdots f_1(\theta)\cdots)\right) \quad \text{with} \quad f_n(\theta) := \mathbb{E}_{\xi_n}\left[f_n(\theta;\xi_n)\right] \tag{2.1}$$

where $\theta \in \mathbb{R}^d$ is the optimization variable, $f_n : \mathbb{R}^{d_n} \to \mathbb{R}^{d_{n+1}}, n = 1, 2, \ldots, N$ (with $d_{N+1} = 1$ and $d_1 = d$) are smooth but possibly nonconvex functions, and $\xi_1, \ldots, \xi_N$ are independent random variables. The formulation (2.1) covers a broader range of applications than the classical non-compositional stochastic optimization and the empirical risk minimization problem in machine learning, e.g., [7]. In the non-compositional cases, the problem is to solve $\min_{\theta\in\mathbb{R}^d} \mathbb{E}_\xi\left[f(\theta;\xi)\right]$, which can be formulated under (2.1) when $f_1(\theta)$ is a scalar function and $f_2, \cdots, f_N$ are the scalar identity maps, e.g., $d_{N+1} = d_N = \cdots = d_2 = 1$ and $d_1 = d$.

Problem (2.1) naturally arises in a number of other areas. In reinforcement learning, finding the value function of a given policy (often referred to as *policy evaluation*) can be casted as a compositional optimization problem; see e.g., [16, 117]. In financial engineering, the risk-averse portfolio optimization can be also formulated in similar form [101]. A recent application of (2.1) is the *model-agnostic meta learning* (MAML), which is under a broader concept of few-shot meta learning; see e.g., [26]. It is a powerful tool for learning a new task by using the prior experience from related tasks. Consider a set of empirically observed tasks collected in $\mathcal{M} := \{1, \ldots, M\}$ drawn from a certain task distribution. By a slight abuse of notation, each task m has its local data ξ_m from a certain distribution, which defines its loss function as $F_m(\theta) := \mathbb{E}_{\xi_m}[f(\theta; \xi_m)]$, $m \in \mathcal{M}$, where $\theta \in \mathbb{R}^d$ is the parameter of a prediction model (e.g., weights in a neural network), and $f(\theta; \xi_n)$ is the individual loss with respect to each datum. In MAML, the goal is to find a common initialization that can adapt to a desired model for a set of new tasks after taking several gradient descent steps. Specifically, we find such initialization by solving the following empirical version of *one-step* MAML problem

$$\min_{\theta \in \mathbb{R}^d} F(\theta) := \frac{1}{M} \sum_{m=1}^{M} F_m(\theta - \alpha \nabla F_m(\theta)) \tag{2.2}$$
$$\text{with} \quad F_m(\theta) := \mathbb{E}_{\xi_m}[f(\theta; \xi_m)]$$

where α is the stepsize, and ∇F_m is the gradient of the loss function at task m. The problem (2.2) is called the one-step adaptation since the loss of each task is evaluated at the model $\theta - \alpha \nabla F_m(\theta)$ that is updated by taking one gradient descent of the each task's loss function. It is not hard to verify that (2.2) can be formulated as the special case of (2.1) with $N = 2$.

Despite its generality and importance, stochastic compositional optimization in the form of (2.1) is not fully explored, especially compared with the major efforts that have been taken for its non-compositional counterpart during the last decade. Averaging, acceleration, and variance reduction are all powerful techniques designed for the non-compositional stochastic optimization. A natural question is *Can we develop a simple yet efficient counterpart of SGD for stochastic compositional optimization?* By *simplicity*, we mean the new algorithm has easy-to-implement update without double loop, accuracy-dependent stepsizes, nor increasing batch sizes, and can be easily augmented with existing techniques for improving SGD. By *efficiency*, we mean the new algorithm can achieve the same convergence rate or the gradient query complexity as SGD for stochastic non-compositional problems. This chapter aims to provide an affirmative answer for this question.

2.1.1 Prior art

We review prior contributions that we group in the following categories.

Stochastic compositional optimization. Non-asymptotic analysis of stochastic compositional optimization is pioneered by [117], where a new approach called SCGD uses two sequences of stepsizes in different time scales: a slower one for updating variable θ, and a faster one for tracking the value of inner function(s). An accelerated variant of SCGD with improved convergence rate has been developed in [118]. In concurrent with our work, an adaptive and accelerated SCGD has been studied in [111], but the updates of [118, 111] are different from ours, and thus their convergence rates are still slower than ours and that of SGD for the non-compositional case. While most of existing algorithms for stochastic compositional problems rely on two-timescale stepsizes, the single timescale approach has been recently developed for the two-level compositional problems in [32], which has been recently extended to the multi-level compositional problems in [94]. Our improvements over [32, 94] are: i) a different and simpler algorithm that tracks only two sequences instead of three; ii) a neat ODE analysis backing up our algorithm development, which may stimulate future development; and, more importantly, iii) the simplicity of both our algorithm makes it easy to adopt the Adam update. In addition, no convergence rate has been established in [94] neither in terms of the gradient norm nor the function values.

Starting from [62], much attention has been paid to a special class of the stochastic compositional problem (2.1) with the *finite-sum structure*. Building upon variance-reduction techniques for non-compositional problems [46, 18, 81, 25], variance-reduced SCGD methods have been developed in this setting under the convex [62, 5, 21, 65], and nonconvex assumptions [39]. Recent advances also include stochastic compositional optimization with a nonsmooth regularizer [40, 130, 131]. Other variants using ADMM and accelerated variance reduction methods for finite-sum compositional problems have been studied in [127, 124]. These variance reduction-based methods have impressive performance in the finite-sum compositional problems. While they can be applied to the stochastic compositional problems (2.1), they require an *increasing batch size* and run in a double-loop manner, which is not preferable in practice.

Optimization for model-agnostic meta learning. On the other end of the spectrum, MAML is a popular framework that learns a good initialization from past experiences for fast adaptation to new tasks [26, 27]. MAML has been applied to various domains including reinforcement learning [67], recommender systems, and communication [102]. Due to the specific formulation, solving MAML requires information on the stochastic Hessian matrix, which can be costly in practice. Some recent efforts have been made towards developing Hessian-free methods for MAML; see also e.g., [82, 23, 54, 103, 24, 90, 134]. While most of existing works aim to find the initialization for the one-step gradient adaptation, the general multi-step MAML has also been recently studied in [43] with improved empirical performance. However, these methods do not fully embrace the compositional structure of MAML, and thus either lead to suboptimal sample complexity or only obtain inexact convergence for (2.2). While this chapter does not deal with Hessian-free update, our algorithms can friendly incorporate these advanced techniques motivated by application-specific challenges as well.

2.1.2 Our contributions

In this context, the present paper puts forward a new stochastic compositional gradient framework that introduces a stochastic correction to the original stochastic compositional gradient method [117], which justifies its name **S**tochastically **C**orrected **S**tochastic **C**ompositional gradient (**SCSC**). Compared to the existing stochastic optimization schemes, our contributions can be summarized as follows.

c1) We develop a stochastic gradient method termed SCSC for stochastic compositional optimization by using stochastically corrected compositional gradients. SCSC is simple to use as its alternatives, yet it achieves the same order of convergence rate $\mathcal{O}(k^{-\frac{1}{2}})$ as SGD for non-compositional problems;

c2) We generalize our SCSC algorithm to solve the multi-level stochastic compositional problems, and develop its adaptive gradient schemes based on the Adam-type update, both of which achieve the same order of convergence rate as their counterparts for non-compositional problems; and,

c3) We empirically verify the effectiveness of our SCSC-based algorithms in the portfolio management and MAML tasks using standard datasets. Comparing with the existing algorithms, our new algorithms converge faster and require a fixed batch size.

2.2 A New Method for Stochastic Compositional Optimization

2.2.1 Warm up: Two-level compositional problems

For the notational brevity, we first consider a special case of (2.1) - the two-level stochastic compositional problem

$$\min_{\theta\in\mathbb{R}^d} \; f(g(\theta)) = \mathbb{E}_\xi \left[f\left(\mathbb{E}_\phi[g(\theta;\phi)];\xi\right)\right] \tag{2.3}$$

where ξ and ϕ are independent random variables. Connecting the notations of (2.3) with those in (2.1), they are $f_2(\,\cdot\,;\xi_2) := f(\,\cdot\,;\xi)$ and $f_1(\theta;\xi_1) := g(\theta;\phi)$.

Before introducing our approach, we first highlight the inherent challenge of applying the standard SGD method to (2.1).

When the distributions of ϕ and ξ are unknown, the stochastic approximation [93] leads to the following stochastic update

$$\theta^{k+1} = \theta^k - \alpha\nabla g(\theta^k;\phi^k)\nabla f(\mathbb{E}_\phi[g(\theta^k;\phi)];\xi^k) \tag{2.4}$$

where ϕ^k and ξ^k are samples drawn at iteration k. Notice that obtaining the unbiased stochastic gradient $\nabla g(\theta^k;\phi^k)\nabla f(\mathbb{E}_\phi[g(\theta^k;\phi)];\xi^k)$ is still costly since the gradient ∇f is evaluated at $\mathbb{E}_\phi[g(\theta^k;\phi)]$. Except that the gradient ∇f is linear, the expectation in (2.4)

Algorithm 1 SCSC for two-level problem

```
initialize: $\theta^0$, $y^0$, stepsizes $\alpha_0, \beta_0$
for $k = 1, 2, \ldots, K$ do
    randomly select datum $\phi^k$
    compute $g(\theta^k; \phi^k)$ and $\nabla g(\theta^k; \phi^k)$
    update variable $y^{k+1}$ via (2.7b) or (2.7c)
    randomly select datum $\xi^k$
    compute $\nabla f(y^{k+1}; \xi^k)$
    update variable $\theta^{k+1}$ via (2.7a)
end for
```

cannot be omitted, because the stochastic gradient $\nabla g(\theta^k; \phi^k)\nabla f(g(\theta^k; \phi^k); \xi^k)$ is biased, i.e.,

$$\mathbb{E}_{\phi^k,\xi^k}[\nabla g(\theta^k; \phi^k)\nabla f(g(\theta^k; \phi^k); \xi^k)] \neq \mathbb{E}_{\phi,\xi}\left[\nabla g(\theta^k; \phi)\nabla f(\mathbb{E}_\phi[g(\theta^k; \phi)]; \xi)\right]. \tag{2.5}$$

Therefore, the machinery of stochastic gradient descent cannot be directly applied here.

To overcome this difficulty, a popular SCGD has been developed in [117] for solving the two-level stochastic compositional problem (2.3), which is given by

$$y^{k+1} = (1 - \beta_k)y^k + \beta_k g(\theta^k; \phi^k) \tag{2.6a}$$

$$\theta^{k+1} = \theta^k - \alpha_k \nabla g(\theta^k; \phi^k)\nabla f(y^{k+1}; \xi^k) \tag{2.6b}$$

where α_k and β_k are two sequences of decreasing stepsizes. The above recursion involves two iterates, y^k and θ^k, whose updates are coupled with each other. To ensure convergence, SCGD requires y^k to be updated in a timescale asymptotically faster than that of θ^k so that θ^k is relatively static with respect to y^k; i.e., $\lim\limits_{k\to\infty} \alpha_k/\beta_k = 0$. This prevents SCGD from choosing the same stepsize as SGD for the non-compositional stochastic problems, and also results in its *suboptimal convergence rate*. In (2.6a), the iterate y^{k+1} linearly combines y^k and $g(\theta^k; \phi^k)$, where y^k is updated by the outdated iterate θ^{k-1}. We notice that this is the main reason of using a smaller stepsize α_k in the proof of [117].

With more insights given in Section 2.2.2, our new method that we term stochastically corrected stochastic compositional gradient (**SCSC**) addresses this issue by linearly combining a "corrected" version of y^k and $g(\theta^k;\phi^k)$. Roughly speaking, if $y^k \approx g(\theta^{k-1})$, we gauge that $g(\theta^k) \approx g(\theta^{k-1}) + \nabla g(\theta^k;\phi^k)(\theta^k - \theta^{k-1})$. Therefore, we propose the following new update

$$\theta^{k+1} = \theta^k - \alpha_k \nabla g(\theta^k;\phi^k)\nabla f(y^{k+1};\xi^k) \tag{2.7a}$$

$$y^{k+1} = (1-\beta_k)\left(y^k + \nabla g(\theta^k;\phi^k)(\theta^k - \theta^{k-1})\right) + \beta_k g(\theta^k;\phi^k). \tag{2.7b}$$

We can also approximate $\nabla g(\theta^k;\phi^k)(\theta^k - \theta^{k-1})$ by the first-order Taylor expansion, that is

$$y^{k+1} = (1-\beta_k)\left(y^k + g(\theta^k;\phi^k) - g(\theta^{k-1};\phi^k)\right) + \beta_k g(\theta^k;\phi^k). \tag{2.7c}$$

Different from (2.6), we use two sequences of stepsizes α_k and β_k in (2.7) that decrease at the same rate as SGD. As we will show later, under a slightly different assumption, both (2.7b) and (2.7c) can guarantee that the new approach achieves the same convergence rate $\mathcal{O}(k^{-\frac{1}{2}})$ as SGD for the non-compositional stochastic optimization problems. Per iteration, (2.7b) requires the same number of function and gradient evaluations as SCGD, and (2.7c) requires one more evaluation at the old iterate θ^{k-1}.

2.2.2 Algorithm development motivated by ODE analysis.

We provide some intuition of our design via an ODE-based construction for the corresponding deterministic continuous-time system. To achieve so, we make the following assumptions [117, 62, 131].

Assumption 1. *Functions f and g are L_f- and L_g-smooth, that is, for any $\theta, \theta' \in \mathbb{R}^d$, we have* $\|\nabla f(\theta;\xi) - \nabla f(\theta';\xi)\| \leq L_f\|\theta - \theta'\|, \quad \|\nabla g(\theta;\phi) - \nabla g(\theta';\phi)\| \leq L_g\|\theta - \theta'\|$.

Assumption 2. *The stochastic gradients of f and g are bounded in expectation, that is* $\mathbb{E}\left[\|\nabla g(\theta;\phi)\|^2\right] \leq C_g^2$ *and* $\mathbb{E}\left[\|\nabla f(y;\xi)\|^2\right] \leq C_f^2$.

Assumptions 1 and 2 require both the function values and the gradients to be Lipschitz continuous. As a result, the compositional function $F(\theta) = f(g(\theta))$ is also smooth with

$L := C_g^2 L_f + C_f L_g$ [131].

Let t be time in this subsection. Consider the following ODE

$$\dot{\theta}(t) = -\alpha \nabla g(\theta(t)) \nabla f(y(t)) \tag{2.8}$$

where the constant $\alpha > 0$. If we set $y(t) = g(\theta(t))$, then this system describes a gradient flow that monotonically decreases $f(g(\theta(t)))$. In this case, we have $\frac{d}{dt} f(g(\theta(t))) = \langle \nabla g(\theta(t)) \nabla f(g(\theta(t))), \dot{\theta}(t) \rangle = -\frac{1}{\alpha} \|\dot{\theta}(t)\|^2$. However, if we can evaluate gradient ∇f only at $y(t) \neq g(\theta(t))$, it introduces inexactness and thus $f(g(\theta(t)))$ may lose monotonicity, namely

$$\begin{aligned} \frac{d}{dt} f(g(\theta(t))) &\overset{(a)}{=} -\frac{1}{\alpha}\|\dot{\theta}(t)\|^2 + \langle \nabla g(\theta(t)) \big(\nabla f(g(\theta(t))) - \nabla f(y(t)) \big), \dot{\theta}(t) \rangle \\ &\overset{(b)}{\leq} -\frac{1}{\alpha}\|\dot{\theta}(t)\|^2 + \|\nabla g(\theta(t))\| \|\nabla f(g(\theta(t))) - \nabla f(y(t))\| \|\dot{\theta}(t)\| \\ &\overset{(c)}{\leq} -\frac{1}{2\alpha}\|\dot{\theta}(t)\|^2 + \frac{\alpha C_g^2 L_f^2}{2} \|g(\theta(t)) - y(t)\|^2 \end{aligned} \tag{2.9}$$

where (a) follows from (2.8), (b) uses the Cauchy-Schwarz inequality, (c) is due to Assumptions 1 and 2 as well as the Young's inequality. In general, the RHS of (2.9) is not necessarily negative. Therefore, it motivates an energy function with both $f(g(\theta(t)))$ and $\|g(\theta(t)) - y(t)\|^2$, given by

$$\mathcal{V}(t) := f(g(\theta(t))) + \|g(\theta(t)) - y(t)\|^2. \tag{2.10}$$

We wish $\mathcal{V}(t)$ would monotonically decrease. By substituting the bound in (2.9), we have

$$\begin{aligned} \dot{\mathcal{V}}(t) \leq& -\frac{1}{2\alpha}\|\dot{\theta}(t)\|^2 + \frac{\alpha C_g^2 L_f^2}{2}\|g(\theta(t)) - y(t)\|^2 + 2\left\langle y(t) - g(\theta(t)), \dot{y}(t) - \nabla g(\theta(t))\dot{\theta}(t) \right\rangle \\ =& -\frac{1}{2\alpha}\|\dot{\theta}(t)\|^2 - \left(2\beta - \frac{\alpha C_g^2 L_f^2}{2}\right)\|g(\theta(t)) - y(t)\|^2 \\ &+ 2\left\langle y(t) - g(\theta(t)), \dot{y}(t) + \beta(y(t) - g(\theta(t))) - \nabla g(\theta(t))\dot{\theta}(t) \right\rangle \end{aligned} \tag{2.11}$$

where $\beta > 0$ is a fixed constant. The first two terms in the RHS of (2.11) are non-positive given that $\alpha \geq 0$ and $\beta \geq \alpha C_g^2 L_f^2 / 4$, but the last term can be either positive or negative. Following the **maximum descent principle** of $\mathcal{V}(t)$, we are motivated to use the following dynamics

$$\dot{y}(t) = -\beta\left(y(t) - g(\theta(t))\right) + \nabla g(\theta(t))\dot{\theta}(t) \implies \dot{\mathcal{V}}(t) \leq 0. \tag{2.12}$$

Directly implementing (2.12) in the discrete time is intractable. Instead, we approximate the continuous-time update by either the backward difference or the Taylor expansion, given by

$$\nabla g(\theta(t))\dot{\theta}(t) \approx \gamma_k \nabla g(\theta^k)\Big(\theta^k - \theta^{k-1}\Big) \tag{2.13a}$$

$$\text{or} \quad \approx \gamma_k \Big(g(\theta^k) - g(\theta^{k-1})\Big) \tag{2.13b}$$

where k is the discrete iteration index, and $\gamma_k > 0$ is the weight controlling the approximation.

With the insights gained from (2.8) and (2.12), our stochastic update (2.7) essentially discretizes time t into iteration k, and replaces the exact function $g(\theta(t))$ and the gradients $\nabla g(\theta(t)), \nabla f(y(t))$ by their stochastic values. The choice $\gamma_k := 1 - \beta_k$ in (2.13) will simplify some constants in the proof.

Connection to existing approaches. Using this interpretation, the dynamics of $y(t)$ in SCGD [117] is

$$\dot{y}(t) = -\beta\left(y(t) - g(\theta(t))\right) \tag{2.14}$$

which will leave an additional non-negative term $\langle y(t) - g(\theta(t)), -\nabla g(\theta(t))\dot{\theta}(t)\rangle \leq C_g\|y(t) - g(\theta(t))\|\|\dot{\theta}(t)\|$ in (2.11). To ensure the convergence of $\mathcal{V}(t)$, a much smaller stepsize α is needed.

Using the ODE interpretation, the dynamics of $y(t)$ in the recent *variance-reduced* compositional gradient approaches, e.g., [62, 39, 130, 131] can be written as

$$\dot{y}(t) = \nabla g(\theta(t))\dot{\theta}(t) \tag{2.15}$$

which leaves the non-negative term $\|g(\theta(t)) - y(t)\|^2$ uncancelled in (2.11). Therefore, to ensure convergence of $\mathcal{V}(t)$, the variance-reduced compositional approaches must calculate the *full gradient* $\nabla f(g(\theta(t)))$ periodically to erase the error accumulated by $\|g(\theta(t)) - y(t)\|^2$.

2.3 Adam-type and Multi-level Variants

In this section, we introduce two variants of our new stochastic compositional gradient method: adaptive stochastic gradient and multi-level compositional gradient schemes.

2.3.1 Adam-type adaptive gradient approach

When the sought parameter θ represents the weight of a neural network, in the non-compositional stochastic problems, finding a good parameter θ will be much more efficient if adaptive SGD approaches are used such as AdaGrad [22] and Adam [55]. We first show that our SCSC method can readily incorporate Adam update for θ, and establish that it achieves the same convergence rate as the original Adam approach for the non-compositional stochastic problems [97, 13].

Following the Adam and its modified approach AMSGrad in [55, 97, 13], the Adam SCSC approach uses two sequences h^k and v^k to track the exponentially weighted gradient of θ^k and its second moment estimates, and uses v^k to inversely weight the gradient estimate h^k. The update can be written as

$$h^{k+1} = \eta_1 h^k + (1-\eta_1)\boldsymbol{\nabla}^k \tag{2.16a}$$

$$v^{k+1} = \eta_2 \hat{v}^k + (1-\eta_2)(\boldsymbol{\nabla}^k)^2 \tag{2.16b}$$

$$\theta^{k+1} = \theta^k - \alpha_k \frac{h^{k+1}}{\sqrt{\epsilon + \hat{v}^{k+1}}} \tag{2.16c}$$

$$y^{k+1} \text{ via (2.7b) or (2.7c)}$$

where the gradient is defined as $\boldsymbol{\nabla}^k := \nabla g(\theta^k;\phi^k)\nabla f(y^{k+1};\xi^k)$; $\hat{v}^{k+1} := \max\{v^{k+1}, \hat{v}^k\}$ ensures the monotonicity of the scaling factor in (2.16c); the constant vector is $\epsilon > 0$; and η_1 and η_2 are two exponential weighting parameters. The vector division and square in (2.16) are defined element-wisely.

The key difference of the Adam-SCSC relative to the original Adam is that the stochastic gradient $\boldsymbol{\nabla}^k$ used in the updates (2.16a) and (2.16b) is not an unbiased estimate of the true one $\nabla F(\theta^k)$. Furthermore, the gradient bias incurred by the Adam update intricately depends on the multi-level compositional gradient estimator, the analysis of which is not only challenging but also of its independent interest.

2.3.2 Multi-level compositional problems

Aiming to solve practical problems with more general stochastic compositional structures, we extend our SCSC method in Section 2.2 for (2.3) to the multi-level problem (2.1). As an example, the *multi-step* MAML problem [43] can be formulated as the multi-level compositional problem (2.1). In this case, a globally shared initial model θ for the N-step adaptation can be found by solving

$$\min_{\theta \in \mathbb{R}^d} \; F(\theta) := \frac{1}{M} \sum_{m=1}^{M} F_m\left(\tilde{\theta}_m^N(\theta)\right) \tag{2.17}$$
$$\text{with} \quad \tilde{\theta}_m^{n+1} = \tilde{\theta}_m^n - \alpha \nabla F_m(\tilde{\theta}_m^n) \quad \text{recursively}$$

where $\tilde{\theta}_m^N(\theta)$ is obtained after N step gradient descent on task m and initialized with $\tilde{\theta}_m^0 = \theta$.

Different from SCSC for the two-level compositional problem (2.3), the multi-level SCSC (**multi-SCSC**) requires to track $N-1$ functions $f_1, \cdots, f_{N-1}$ using $y_1, \cdots, y_{N-1}$. Following the tracking update of SCSC, the multi-SCSC update is

$$y_1^{k+1} = (1-\beta_k) y_1^k + \beta_k f_1(\theta^k; \xi_1^k) + (1-\beta_k)(f_1(\theta^k; \xi_1^k) - f_1(\theta^{k-1}; \xi_1^k)) \tag{2.18a}$$

$$\cdots$$

$$\begin{aligned} y_{N-1}^{k+1} = {} & (1-\beta_k) y_{N-1}^k + \beta_k f_{N-1}(y_{N-2}^{k+1}; \xi_{N-1}^k) \\ & + (1-\beta_k)(f_{N-1}(y_{N-2}^{k+1}; \xi_{N-1}^k) - f_{N-1}(y_{N-2}^k; \xi_{N-1}^k)) \end{aligned} \tag{2.18b}$$

$$\theta^{k+1} = \theta^k - \alpha_k \nabla f_1(\theta^k; \xi_1^k) \cdots \nabla f_{N-1}(y_{N-2}^{k+1}; \xi_{N-1}^k) \nabla f_N(y_{N-1}^{k+1}; \xi_N^k). \tag{2.18c}$$

Note that both (2.7b) and (2.7c) can be used in multi-SCSC (2.18), though above we choose (2.7c). Multi-SCSC can also incorporate Adam-type update. Analyzing multi-SCSC is more challenging that SCSC, since the tracking variables are statistically dependent on each other.

Algorithm 2 Adam SCSC method

```
**initialize:** $\theta^0$, $y^0$, $v^0$, $h^0$, $\eta_1$, $\eta_2$, $\alpha_0, \beta_0$
**for** $k = 1, 2, \ldots, K$ **do**
    randomly select datum $\phi^k$
    compute $g(\theta^k; \phi^k)$ and $\nabla g(\theta^k; \phi^k)$
    update variable $y^{k+1}$ via (2.7b) or (2.7c)
    randomly select datum $\xi^k$
    compute $\nabla f(y^{k+1}; \xi^k)$
    update $h^{k+1}, v^{k+1}, \theta^{k+1}$ via (2.16)
**end for**
```

Specifically, conditioned on the randomness up to iteration k, the variable y_n^{k+1} depends on y_{n-1}^{k+1} and thus also on $y_{n-2}^{k+1}, \cdots, y_1^{k+1}$. Albeit its complex compositional form, as we will shown later, multi-SCSC also achieves the same rate of convergence as SGD for non-compositional stochastic optimization.

2.4 Convergence Analysis of SCSC

In this section, we establish the convergence of all SCSC algorithms. For our analysis, in addition to Assumptions 1 and 2, we make the following assumptions.

Assumption 3. *Random sampling oracle satisfies that* $\mathbb{E}\left[g(\theta;\phi^k)\right] = g(\theta)$*, and* $\mathbb{E}\big[\nabla g(\theta;\phi^k)$ $\nabla f(y;\xi^k)\big] = \nabla g(\theta)\nabla f(y)$*.*

Assumption 4. *Function* $g(\theta;\phi^k)$ *has bounded variance, i.e.,* $\mathbb{E}\left[\|g(\theta;\phi^k) - g(\theta)\|^2\right] \leq V_g^2$*.*

Assumptions 3 and 4 are standard in stochastic compositional optimization; e.g., [117, 118, 62, 131], and are analogous to the unbiasedness and bounded variance assumptions for non-compositional problems.

2.4.1 Convergence in the two-level case

With insights gained from the continuous-time Lyapunov function (2.10), our analysis in this subsection critically builds on the following discrete-time Lyapunov function:

$$\mathcal{V}^k := F(\theta^k) - F(\theta^*) + \|g(\theta^{k-1}) - y^k\|^2 \tag{2.19}$$

where θ^* is the optimal solution of the problem (2.3).

Lemma 1 (Tracking variance of SCSC) *Consider $\mathcal{F}^k$ as the collection of random variables, i.e., $\mathcal{F}^k := \left\{\phi^0, \ldots, \phi^{k-1}, \xi^0, \ldots, \xi^{k-1}\right\}$. Suppose Assumptions 1-4 hold, and y^{k+1} is generated by running SCSC iteration* (2.7a) *and* (2.7c) *conditioned $\mathcal{F}^k$. The mean square error of y^{k+1} satisfies*

$$\mathbb{E}\left[\|g(\theta^k) - y^{k+1}\|^2 \mid \mathcal{F}^k\right] \leq (1-\beta_k)^2\|g(\theta^{k-1}) - y^k\|^2 + 4(1-\beta_k)^2 C_g^2\|\theta^k - \theta^{k-1}\|^2 + 2\beta_k^2 V_g^2. \tag{2.20}$$

Intuitively, since $\|\theta^k - \theta^{k-1}\|^2 = \mathcal{O}(\alpha_{k-1}^2)$, Lemma 1 implies that if the stepsizes α_k^2 and β_k^2 are decreasing, the mean square error of y^{k+1} will decrease. Note that Lemma 1 presents the performance of y^{k+1} using the update (2.7c). If we use the update (2.7b) instead, the bound in (2.20) will have an additional term $(1-\beta_k)^2\|\theta^k - \theta^{k-1}\|^4$. Under a stronger version of Assumption 2 (e.g., fourth moments), the remaining analysis still follows; see the derivations in supplementary material.

Building upon Lemma 1, we establish the following theorem.

Theorem 1 (two-level SCSC) *Under Assumptions 1-4, if we choose the stepsizes as $\alpha_k = \frac{2\beta_k}{C_g^2 L_f^2} = \alpha = \frac{1}{\sqrt{K}}$, the iterates $\{\theta^k\}$ of SCSC in Algorithm 1 satisfy*

$$\frac{\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla F(\theta^k)\|^2]}{K} \leq \frac{2\mathcal{V}^0 + 2B_1}{\sqrt{K}} \tag{2.21}$$

where the constant is defined as $B_1 := \frac{L}{2}C_g^2 C_f^2 + 4V_g^2 + 16C_g^4 C_f^2$.

Theorem 1 implies that the convergence rate of SCSC is $\mathcal{O}(k^{-\frac{1}{2}})$, which is on the same order of SGD's convergence rate for the stochastic non-compositional nonconvex problems [29], and significantly improves $\mathcal{O}(k^{-\frac{1}{4}})$ of the original SCGD [117] and $\mathcal{O}(k^{-\frac{4}{9}})$ of its accelerated version [118]. Comparing with [32, 94] that achieves the same rate of $\mathcal{O}(k^{-\frac{1}{2}})$ for the two-level problem, our algorithm is simpler which makes it possible to adopt the Adam update. In addition, this rate is not directly comparable to those under variance-reduced compositional methods, e.g., [62, 39, 130, 131] since SCSC does not need the increasing batchsize nor double-loop.

2.4.2 Convergence of Adam-SCSC

The convergence analysis for Adam SCSC builds on the following Lyapunov function:

$$\mathcal{V}^k := F(\theta^k) - F(\theta^*) - \sum_{j=k}^{\infty} \eta_1^{j-k+1} \alpha_j \left\langle \nabla F(\theta^{k-1}), \frac{h^k}{\sqrt{\epsilon + \hat{v}^k}} \right\rangle + c \left\| g(\theta^{k-1}) - y^k \right\|^2 \quad (2.22)$$

where c is a constant depends on η_1, η_2 and ϵ. Clearly, the Lyapunov function (2.22) is a generalization of (2.19) for SCSC, which takes into account the adaptive gradient update by subtracting the inner product between the full gradient and the Adam SCSC update. Intuitively, if the adaptive stochastic gradient direction is aligned with the gradient direction, this term will also become small.

To establish the convergence of Adam SCSC, we need a slightly stronger version of Assumption 2, which is standard in analyzing the convergence of Adam [55, 97, 13].

Assumption 5. *Stochastic gradients are bounded almost surely,* $\|\nabla g(\theta; \phi)\| \leq C_g$, $\|\nabla f(y; \xi)\| \leq C_f$. Analogous to Theorem 1, we establish the convergence of Adam SCSC under nonconvex settings.

Theorem 2 (Adam SCSC) *Under Assumptions 1 and 3-5, if we choose the parameters* $\eta_1 < \sqrt{\eta_2} < 1$*, and the stepsizes as* $\alpha_k = 2\beta_k = \frac{1}{\sqrt{K}}$*, the iterates* $\{\theta^k\}$ *of Adam SCSC in*

Algorithm 2 satisfy

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla F(\theta^k)\|^2] \leq \frac{2(\epsilon + C_g^2C_f^2)^{\frac{1}{2}}}{(1-\eta_1)}$$
$$\left(\frac{\mathcal{V}^0 + (4C_g^2\tilde{\eta} + V_g^2)c + 2d\tilde{\eta}L}{\sqrt{K}} + \frac{C_gC_fd\tilde{\eta}}{K} + \frac{(1+(1-\eta_1)^{-1})C_g^2C_f^2d\epsilon^{-\frac{1}{2}}}{K}\right) \tag{2.23}$$

where d is the dimension of θ, and the constant is defined as $\tilde{\eta} := (1-\eta_1)^{-1}(1-\eta_2)^{-1}(1-\eta_1^2/\eta_2)^{-1}$.

Theorem 2 implies that the convergence rate of Adam SCSC is also $\mathcal{O}(k^{-\frac{1}{2}})$. This rate is again on the same order of Adam's convergence rate for the stochastic non-compositional nonconvex problems [13], and significantly faster than $\mathcal{O}(k^{-\frac{4}{9}})$ of the existing adaptive compositional SGD method [111]. As a by-product, the newly designed Lyapunov function (2.22) also significantly streamlines the original analysis of Adam under nonconvex settings [13], which is of its independent interest.

2.4.3 Convergence of multi-SCSC

In this section, we establish the convergence results of the multi-level SCSC, and present the corresponding analysis.

The subsequent analysis for the multi-level problem builds on the following *Lyapunov function*:

$$\mathcal{V}^k := F(\theta^k) - F(\theta^*) + \sum_{n=1}^{N-1}\left\|y_n^k - f_n(y_{n-1}^k)\right\|^2 \tag{2.24}$$

where θ^* is the optimal solution of the problem (2.1).

To this end, we need a generalized version of Assumptions 1-4 for the multi-level setting.

Assumption m1. *Functions $\{f_n\}$ are L_n-smooth, that is, for any $\theta, \theta' \in \mathbb{R}^d$, $\|\nabla f_n(\theta;\xi_n) - \nabla f_n(\theta';\xi_n)\| \leq L_n\|\theta - \theta'\|$.*

Assumption m2. *The stochastic gradients of $\{f_n\}$ are bounded in expectation, that is $\mathbb{E}\left[\|\nabla f_n(\theta;\xi_n)\|^2\right] \leq C_n^2$.*

Assumption m3. *Random sampling oracle satisfies that* $\mathbb{E}\left[f_n(\theta;\xi_n^k)\right] = f_n(\theta)$, $\forall n$, *and* $\mathbb{E}\left[\nabla f_1(\theta;\xi_1^k)\cdots\nabla f_N(y_{N-1};\xi_N^k)\right] = \nabla f_1(\theta)\cdots\nabla f_N(y_{N-1})$.

Assumption m4. *For all* n, $f_n(\theta;\xi_n)$ *has bounded variance, i.e.,* $\mathbb{E}\left[\|f_n(\theta;\xi_n) - f_n(\theta)\|^2\right] \leq V^2$.

Building upon these assumptions, we establish the convergence of multi-SCSC under nonconvex settings.

Theorem 3 (multi-level SCSC) *Under Assumptions m1-m4, if we choose the stepsizes as* $\alpha_k = \frac{2\beta_k}{\sum_{n=1}^{N-1} A_n^2} = \frac{1}{\sqrt{K}}$, *the iterates* $\{\theta^k\}$ *of the multi-level SCSC iteration* (2.18) *satisfy*

$$\frac{\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla F(\theta^k)\|^2]}{K} \leq \frac{2\mathcal{V}^0 + 2(B_2 + \tilde{B}_2(\sum_{n=1}^{N-1} A_n^2)^2/4)}{\sqrt{K}}. \tag{2.25}$$

where $B_2, B_3, A_1, \ldots, A_N$ *are some constants that depend on* $C_1, \ldots, C_N$ *and* $L_1, \ldots, L_N$.

Theorem 3 implies that the convergence rate of multi-SCSC is also $\mathcal{O}(k^{-\frac{1}{2}})$. This rate is again on the same order of SGD's rate for the stochastic non-compositional nonconvex problems.

2.5 Numerical Tests

To validate our theoretical results, this section evaluates the empirical performance of our SCSC and Adam SCSC. We evaluate the empirical performance of SCSC and Adam SCSC in two tasks: **sinusoidal regression for MAML** and **risk-averse portfolio management**. All experiments are run on a computer with Intel i9-9960x and NVIDIA Titan GPU.

2.5.1 Sinusoidal regression for MAML

For MAML, we consider the sinusoidal regression tasks as that in [26]. Each task in MAML is to regress from the input to the output of a sine wave

$$s(x;a,\varphi) = a\sin(x+\varphi) \tag{2.26}$$

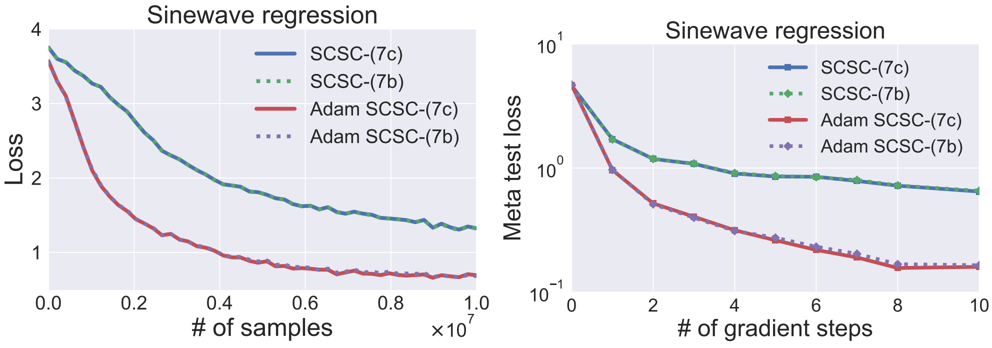

Figure 2.1: Comparison of two SCSC updates on the Sinewave regression task.

where the amplitude a and phase φ of the sinusoid vary across tasks. We sample the amplitude a uniformly from $\mathcal{U}([0.1, 5])$ and the phase φ uniformly from $\mathcal{U}([0, 2\pi])$. During training, datum x is sampled uniformly from $\mathcal{U}([-5, 5])$ and $s(x; a, \varphi)$ is observed. We use a neural network with 2 layers of hidden neurons with weights θ as the regressor $\hat{s}(x; \theta)$ and use the mean square error $\mathbb{E}_x[\|\hat{s}(x; \theta) - s(x; a, \varphi)\|^2]$. We define

$$F_m(\theta) = \mathbb{E}_x[\|\hat{s}(x; \theta) - s(x; a_m, \varphi_m)\|^2]. \tag{2.27}$$

In this case, to connect with (2.3), both random variables ξ and ϕ in (2.3) are uniformly drawn from $\mathcal{U}([-5, 5])$. Let us define

$$\begin{aligned} g(\theta) &= [g_1(\theta)^\top, \cdots, g_M(\theta)^\top]^\top \\ &= [(\theta - \nabla F_1(\theta))^\top, \cdots, (\theta - \nabla F_M(\theta))^\top]^\top \in \mathbb{R}^{Md} \end{aligned} \tag{2.28}$$

and define $y_m \in \mathbb{R}^d$ to track $g_m(\theta)$. With $y := [y_1^\top, \cdots, y_M^\top]^\top \in \mathbb{R}^{Md}$, we define

$$f(y) := \frac{1}{M} \sum_{m=1}^{M} F_m(y_m). \tag{2.29}$$

Then MAML with sinusoidal regression satisfies the composition formulation (2.3).

Benchmark algorithms. In Figure 2.1, we first compare the performance of SCSC and Adam SCSC under two different rules (2.7b) and (2.7c). We then compare our SCSC and Adam SCSC with non-compositional stochastic optimization solver Adam and SGD (the

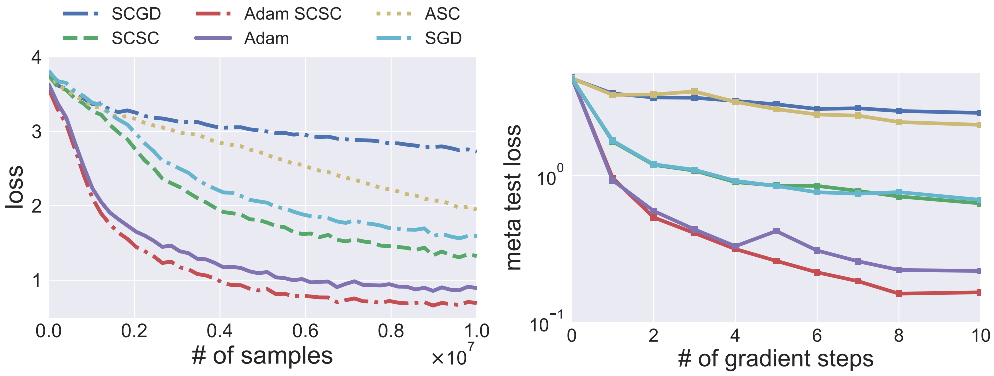

Figure 2.2: Summary of results on the Sinewave regression task.

common baseline for MAML), as well as compositional stochastic solver SCGD and ASC in Figure 2.2.

Hyperparameter tuning. We tune the hyperparameters by first following the suggested order of stepsizes from the original papers and then using a grid search for the constant. For SCSC and Adam SCSC, we use stepsizes $\alpha, \beta_k = 0.8$. For Adam and SGD, we use α. For SCGD and ASC, we use stepsizes $\alpha_k = \alpha k^{-3/4}, \beta_k = k^{-1/2}$ and $\alpha_k = \alpha k^{-5/9}$ and $\beta_k = k^{-4/9}$ as suggested in [117, 118]. The initial learning rate α is chosen from $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ and optimized for each algorithm.

During training, we fix $M = 100$ and we sample 10 data from each task to evaluate the inner function $g(\theta)$, and use another 10 data to evaluate $f(y)$. The MAML adaptation stepsize in (2.2) is $\alpha = 0.01$.

We compare the performance of SCSC and Adam SCSC under two different update rules (2.7b) and (2.7c) in Figure 2.1 for the sinewave regression MAML task. Both (2.7b) and (2.7c) can guarantee that the new approach achieves the same convergence rate $\mathcal{O}(k^{-\frac{1}{2}})$, but (2.7c) requires one more function evaluation than (2.7b) at the old iterate θ^{k-1}. In terms of both the number of samples and number of gradients, two update rules have very close performance, and the two lines are almost overlapping.

In Figure 2.2, at each evaluation point of test loss, we sample 100 data to test the

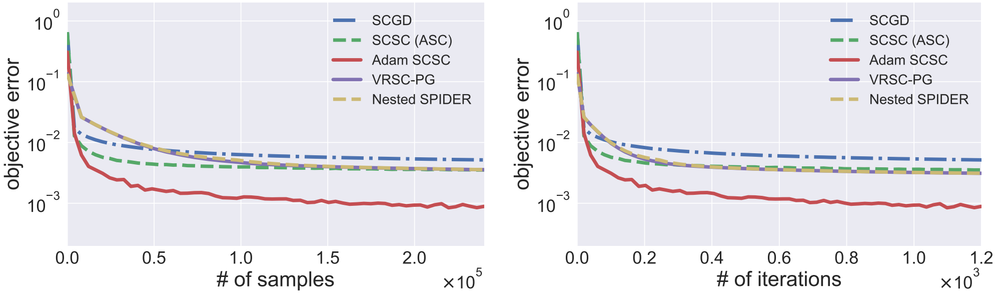

Figure 2.3: Summary of results on the *Industrial-49* dataset.

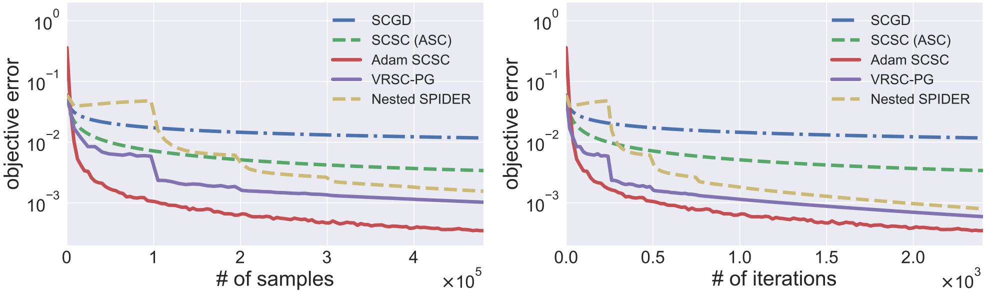

Figure 2.4: Summary of results on the *100 Book-to-Market* dataset.

performance of each algorithm on these trained tasks. We also sample 100 unseen tasks to test the adaptation of the meta parameter learned on $M = 100$ tasks. For each unseen task, we start with the learned initialization and perform 10-step SGD with minibatch of 10. As shown in Figure 2.2, in terms of training loss, Adam SCSC again achieves the best performance, and SCSC outperforms the popular SCGD and ASC methods. In the meta test, while all algorithms reduce the test loss after several steps of adaptation, Adam SCSC achieves the fastest adaptation, and SCSC also has competitive performance.

2.5.2 Risk-averse portfolio management

Given d assets, let $r_t \in \mathbb{R}^d$ denote the reward vector with nth entry representing the reward of nth asset observed at time slot t over a total of T slots. Portfolio management aims to find an investment $\theta \in \mathbb{R}^d$ with nth entry representing the amount of investment or the split of the total investment allocated to the asset n. The optimal investment θ^* is the one that solves the following problem

$$\max_{\theta\in\mathbb{R}^d} \frac{1}{T}\sum_{t=1}^{T} r_t^\top\theta - \frac{1}{T}\sum_{t=1}^{T}\left(r_t^\top\theta - \frac{1}{T}\sum_{j=1}^{T} r_j^\top\theta\right)^2. \tag{2.30}$$

In this case, both random variables ξ and ϕ in (2.3) are uniformly drawn from $\{r_1, \cdots, r_T\}$. If we define $g(\theta; r_j) = [\theta, r_j^\top\theta]^\top \in \mathbb{R}^{d+1}$, and $y \in \mathbb{R}^{d+1}$ tracking $\mathbb{E}[g(\theta; r)]$, and define

$$f(y; r_t) = y_{(d+1)} - (y_{(d+1)} - r_t^\top y_{(1:d)})^2 \tag{2.31}$$

where $y_{(1:d)}$ and $y_{(d+1)}$ denote the first d entries and the $(d+1)$th entry of y. In this case, problem (2.30) is an instance of stochastic composition problem (2.3).

Benchmark algorithms. We compare SCSC and Adam SCSC with SCGD[117], VRSC-PG [40] and Nested SPIDER [131]. For linear $g(\theta; r)$, it can be verified that SCSC is equivalent to the accelerated SCGD (ASC) [118], and our SCSC and Adam SCSC under two different inner update rules (2.7b) and (2.7c) are also equivalent. Therefore, we only include one.

Hyperparameter tuning. We tune the hyperparameters by first following the suggested order of stepsizes from the original papers and then using a grid search for the constant. For example, we choose $\alpha_k = \alpha k^{-3/4}, \beta_k = k^{-1/2}$ for SCGD; $\alpha_k = \alpha k^{-1/2}, \beta_k = k^{-1/2}$ for SCSC and Adam SCSC; the constant stepsize α for VRSC-PG and Nested SPIDER. The initial learning rate α is chosen from the searching grid $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ and optimized for each algorithm in terms of loss versus the number of iterations. Note that whenever the best performing hyperparameter lies in the boundary of the searching grid, we always extend the grid to make the final hyperparameter fall into the interior of the grid. For all the algorithms, we use the batch size 100 for both inner and outer functions. Figures 2.3 and 2.4 show the test results averaged over 50 runs on two benchmark datasets: *Industrial-49* and *100 Book-to-Market*. The two datasets are downloaded from the Keneth R. French Data Library[1]

[1] http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

without preprocessing. On both datasets, Adam SCSC achieves the best performance, and SCSC outperforms several popular alternatives.

2.6 Proofs of results for the two-level SCSC

In this section, we present the proofs of the theorems in Section 2.4. Due to space limitation, we leave the proofs of the multi-level case in the next chapter.

2.6.1 Proof of Theorem 1

2.6.1.1 Proof of Lemma 1 under Option 1

From the update (2.7b), we have that

$$\begin{aligned} y^{k+1} - g(\theta^k) &= (1-\beta_k)(y^k - g(\theta^{k-1})) + (1-\beta_k)(g(\theta^{k-1}) - g(\theta^k)) \\ &\quad + \beta_k(g(\theta^k;\phi^k) - g(\theta^k)) + (1-\beta_k)(g(\theta^k;\phi^k) - g(\theta^{k-1};\phi^k)) \\ &= (1-\beta_k)(y^k - g(\theta^{k-1})) + (1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3. \end{aligned} \tag{2.32}$$

where we define the three terms as $T_1 := g(\theta^{k-1}) - g(\theta^k)$, $T_2 := g(\theta^k;\phi^k) - g(\theta^k)$, and $T_3 := g(\theta^k;\phi^k) - g(\theta^{k-1};\phi^k)$.

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k, we have

$$\mathbb{E}\left[(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3|\mathcal{F}^k\right] = \mathbf{0}. \tag{2.33}$$

Therefore, conditioned on $\mathcal{F}^k$, taking expectation on the both sides of (2.32), we have

$$\begin{aligned} \mathbb{E}[\|y^{k+1} - g(\theta^k)\|^2|\mathcal{F}^k] &= \mathbb{E}[\|(1-\beta_k)(y^k - g(\theta^{k-1}))\|^2|\mathcal{F}^k] \\ &\quad + \mathbb{E}\left[\|(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3\|^2|\mathcal{F}^k\right] \\ &\quad + 2\mathbb{E}\Big[\Big\langle(1-\beta_k)(y^k - g(\theta^{k-1})), (1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3\Big\rangle|\mathcal{F}^k\Big]. \end{aligned}$$

Using the Young's inequality, we have

$$
\begin{aligned}
&\mathbb{E}\left[\|(1-\beta_k)T_1+\beta_k T_2+(1-\beta_k)T_3\|^2|\mathcal{F}^k\right]\\
\le\ & 2\mathbb{E}\left[\|(1-\beta_k)T_1+\beta_k T_2\|^2|\mathcal{F}^k\right]+2(1-\beta_k)^2\mathbb{E}\left[\|T_3\|^2|\mathcal{F}^k\right]\\
\le\ & 2(1-\beta_k)^2\mathbb{E}[\|T_1\|^2\mid\mathcal{F}^k]+2\beta_k^2\mathbb{E}[\|T_2\|^2\mid\mathcal{F}^k]\\
&+4\beta_k(1-\beta_k)\left\langle T_1,\mathbb{E}[T_2\mid\mathcal{F}^k]\right\rangle+2(1-\beta_k)^2\mathbb{E}\left[\|T_3\|^2|\mathcal{F}^k\right]\\
\le\ & 2(1-\beta_k)^2\mathbb{E}\left[\|g(\theta^k)-g(\theta^{k-1})\|^2|\mathcal{F}^k\right]\\
&+2(1-\beta_k)^2\mathbb{E}\left[\|g(\theta^k;\phi^k)-g(\theta^{k-1};\phi^k)\|^2|\mathcal{F}^k\right]+2\beta_k^2V_g^2\\
\le\ & 4(1-\beta_k)^2C_g^2\|\theta^k-\theta^{k-1}\|^2+2\beta_k^2V_g^2
\end{aligned}
$$

from which the proof is complete.

2.6.1.2 Proof of Lemma 1 under Option 2

Lemma 2 (Tracking error under Option 2) *Suppose that Assumptions 1-4 hold, and y^{k+1} is generated by running iteration* (2.7) *given θ^k. Then the variance of y^{k+1} satisfies*

$$
\begin{aligned}
\mathbb{E}\left[\|g(\theta^k)-y^{k+1}\|^2\mid\mathcal{F}^k\right]\le(1-\beta_k)\|y^{k-1}-g(\theta^k)\|^2+4(1-\beta_k)^2C_g^2\|\theta^k-\theta^{k-1}\|^2\\
+2\beta_k^2V_g^2+\frac{(1-\beta_k)^2L^2}{4}\|\theta^k-\theta^{k-1}\|^4. \qquad (2.34)
\end{aligned}
$$

Compared with the tracking variance in Lemma 1 under (2.7c), Lemma 2 under (2.7b) has an additional term $\frac{(1-\beta_k)^2L^2}{4}\|\theta^k-\theta^{k-1}\|^4$. In this case, under a stronger version of Assumption 2' (e.g., bounded fourth moments), this term is $\mathcal{O}\left(\alpha_k^4\right)$, which will be dominated by second and the third terms in the RHS of (2.34) since both of them are $\mathcal{O}\left(\alpha_k^2\right)$.

Once we have established this, the remaining proof of SCSC with (2.7b) follows the same line as that of SCSC with (2.7c). For brevity, we only present the proof under Lemma 1, and that under Lemma 2 follows similarly.

Assumption 2'. *The stochastic gradients of f and g are bounded in expectation, that is $\mathbb{E}\left[\|\nabla g(\theta;\phi)\|^4\right]\le C_g^4$ and $\mathbb{E}\left[\|\nabla f(y;\xi)\|^4\right]\le C_f^4$.*

Proof: For (2.7b), using the fact that $\nabla g(\theta)$ is L_g-Lipchitz continuous, we have

$$\begin{aligned} y^{k+1} - g(\theta^k) =& (1-\beta_k)(y^k - g(\theta^{k-1})) + (1-\beta_k)(g(\theta^k) - g(\theta^{k-1}) \\ & + \beta_k(g(\theta^k;\phi^k) - g(\theta^k)) + (1-\beta_k)\nabla g(\theta^{k-1};\phi^k)(\theta^k - \theta^{k-1}) \\ =& (1-\beta_k)(y^k - g(\theta^{k-1}) + (1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3 \end{aligned} \tag{2.35}$$

where we define the terms as $T_1 := g(\theta^{k-1}) - g(\theta^k)$, $T_2 := g(\theta^k;\phi^k) - g(\theta^k)$, and $T_3 := \nabla g(\theta^{k-1};\phi^k)(\theta^k - \theta^{k-1})$.

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k, we have

$$\begin{aligned} & \left\|\mathbb{E}\left[(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3 \mid \mathcal{F}^k\right]\right\| \qquad (2.36) \\ =& (1-\beta_k)\left\|g(\theta^{k-1}) - g(\theta^k) + \nabla g(\theta^{k-1})(\theta^k - \theta^{k-1})\right\| \\ =& (1-\beta_k)\left\|\int_0^1 -\nabla g(\theta^{k-1} + t(\theta^k - \theta^{k-1}))(\theta^k - \theta^{k-1})dt + \nabla g(\theta^{k-1})(\theta^k - \theta^{k-1})\right\| \\ \leq& (1-\beta_k)\int_0^1 \left\|\nabla g(\theta^{k-1}) - \nabla g(\theta^{k-1} + t(\theta^k - \theta^{k-1}))\right\| \|\theta^k - \theta^{k-1}\| dt \\ \leq& (1-\beta_k)\int_0^1 L_g t \|\theta^k - \theta^{k-1}\|^2 = \frac{(1-\beta_k)L_g}{2}\|\theta^k - \theta^{k-1}\|^2. \end{aligned}$$

Therefore, conditioned on $\mathcal{F}^k$, taking expectation on both sides of (2.35) over ϕ^k, we have

$$\begin{aligned}
&\mathbb{E}[\|y^{k+1}-g(\theta^k)\|^2 \mid \mathcal{F}^k]\\
=&(1-\beta_k)^2\|y^k-g(\theta^{k-1})\|^2+2\left\langle(1-\beta_k)(y^k-g(\theta^{k-1})), \mathbb{E}\left[(1-\beta_k)T_1+\beta_kT_2+(1-\beta_k)T_3 \,|\, \mathcal{F}^k\right]\right\rangle\\
&+\mathbb{E}[\|(1-\beta_k)T_1+\beta_kT_2+(1-\beta_k)T_3\|^2 \mid \mathcal{F}^k]\\
\overset{(2.36)}{\leq}&(1-\beta_k)^2\|y^k-g(\theta^{k-1})\|^2+2\mathbb{E}[\|(1-\beta_k)T_1+\beta_kT_2)\|^2 \mid \mathcal{F}^k]+2(1-\beta_k)^2\mathbb{E}[\|T_3\|^2 \mid \mathcal{F}^k]\\
&+(1-\beta_k)^2L\|y^k-g(\theta^{k-1})\|\|\theta^k-\theta^{k-1}\|^2\\
\leq&(1-\beta_k)^2\|y^k-g(\theta^{k-1})\|^2+2(1-\beta_k)^2\mathbb{E}[\|T_1\|^2 \mid \mathcal{F}^k]+2\beta_k^2\mathbb{E}[\|T_2\|^2 \mid \mathcal{F}^k]\\
&+4\beta_k(1-\beta_k)\left\langle T_1, \mathbb{E}[T_2 \mid \mathcal{F}^k]\right\rangle+2(1-\beta_k)^2\mathbb{E}[\|T_3\|^2 \mid \mathcal{F}^k]\\
&+(1-\beta_k)^2\beta_k\|y^k-g(\theta^{k-1})\|^2+\frac{(1-\beta_k)^2L^2}{4}\|\theta^k-\theta^{k-1}\|^4\\
\leq&(1-\beta_k)^2(1+\beta_k)\|y^k-g(\theta^{k-1})\|^2+2(1-\beta_k)^2\mathbb{E}\left[\|g(\theta^k)-g(\theta^{k-1})\|^2 \mid \mathcal{F}^k\right]+2\beta_k^2V_g^2\\
&+2(1-\beta_k)^2\mathbb{E}\left[\|g(\theta^k;\phi^k)-g(\theta^{k-1};\phi^k)\|^2 \mid \mathcal{F}^k\right]+\frac{(1-\beta_k)^2L^2}{4}\|\theta^k-\theta^{k-1}\|^4\\
\leq&(1-\beta_k)\|y^k-g(\theta^{k-1})\|^2+4(1-\beta_k)^2C_g^2\|\theta^k-\theta^{k-1}\|^2+2\beta_k^2V_g^2+\frac{(1-\beta_k)^2L^2}{4}\|\theta^k-\theta^{k-1}\|^4
\end{aligned}$$

from which the proof is complete.

2.6.1.3 Remaining proof

Using the smoothness of F, we have

$$\begin{aligned}
F(\theta^{k+1})-F(\theta^k)\leq&\langle\nabla F(\theta^k), \theta^{k+1}-\theta^k\rangle+\frac{L}{2}\|\theta^{k+1}-\theta^k\|^2\\
=&-\alpha_k\langle\nabla F(\theta^k), \nabla g(\theta^k;\phi^k)\nabla f(y^{k+1};\xi^k)\rangle+\frac{L}{2}\|\theta^{k+1}-\theta^k\|^2\\
=&-\alpha_k\|\nabla F(\theta^k)\|^2+\frac{L}{2}\|\theta^{k+1}-\theta^k\|^2\\
&+\alpha_k\langle\nabla F(\theta^k), \nabla g(\theta^k)\nabla f(g(\theta^k))-\nabla g(\theta^k;\phi^k)\nabla f(y^{k+1};\xi^k)\rangle.
\end{aligned}$$

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k and ξ^k on both sides, we have

$$
\begin{aligned}
&\mathbb{E}\left[F(\theta^{k+1})|\mathcal{F}^k\right] - F(\theta^k)\\
\overset{(a)}{\leq} & -\alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}\mathbb{E}\left[\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k\right]\\
&+\alpha_k\mathbb{E}\left[\langle\nabla F(\theta^k), \nabla g(\theta^k;\phi^k)(\nabla f(g(\theta^k);\xi^k) - \nabla f(y^{k+1};\xi^k)\rangle|\mathcal{F}^k\right]\\
\overset{(b)}{\leq} & -\alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k]\\
&+\alpha_k\left\|\nabla F(\theta^k)\right\|\mathbb{E}\left[\|\nabla g(\theta^k;\phi^k)\|^2|\mathcal{F}^k\right]^{\frac{1}{2}}\mathbb{E}\left[\|\nabla f(g(\theta^k);\xi^k) - \nabla f(y^{k+1};\xi^k)\|^2|\mathcal{F}^k\right]^{\frac{1}{2}}
\end{aligned}
\tag{2.37}
$$

where (a) uses $\mathbb{E}[\nabla g(\theta^k;\phi^k)\nabla f(g(\theta^k);\xi^k)|\mathcal{F}^k] = \nabla g(\theta^k)\nabla f(g(\theta^k))$ in Assumption 3, and (b) uses the Cauchy-Schwartz inequality.

Further expanding the RHS of (2.37), we have

$$
\frac{L}{2}\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k] \leq \frac{L}{2}C_g^2C_f^2\alpha_k^2 \tag{2.38}
$$

which follows from Assumption 2. And

$$
\begin{aligned}
&\alpha_k\left\|\nabla F(\theta^k)\right\|\mathbb{E}\left[\|\nabla g(\theta^k;\phi^k)\|^2|\mathcal{F}^k\right]^{\frac{1}{2}}\mathbb{E}\left[\|\nabla f(g(\theta^k);\xi^k) - \nabla f(y^{k+1};\xi^k)\|^2|\mathcal{F}^k\right]^{\frac{1}{2}}\\
\overset{(c)}{\leq}&\alpha_k C_g L_f\|\nabla F(\theta^k)\|\mathbb{E}\left[\|g(\theta^k)-y^{k+1}\|^2|\mathcal{F}^k\right]^{\frac{1}{2}}\\
\overset{(d)}{\leq}&\frac{\alpha_k^2}{4\beta_k}C_g^2L_f^2\|\nabla F(\theta^k)\|^2 + \beta_k\mathbb{E}\left[\|g(\theta^k)-y^{k+1}\|^2|\mathcal{F}^k\right]
\end{aligned}
$$

where (c) uses Assumptions 1 and 2; and (d) uses the Young's inequality.

Therefore, we have

$$
\begin{aligned}
&\mathbb{E}\left[F(\theta^{k+1})|\mathcal{F}^k\right] - F(\theta^k)\\
\leq& -\alpha_k\left(1-\frac{\alpha_k}{4\beta_k}C_g^2L_f^2\right)\|\nabla F(\theta^k)\|^2 + \beta_k\mathbb{E}\left[\|g(\theta^k)-y^{k+1}\|^2|\mathcal{F}^k\right] + \frac{L}{2}C_g^2C_f^2\alpha_k^2.
\end{aligned}
\tag{2.39}
$$

Then with the definition of Lyapunov function in (2.19), it follows that

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}|\mathcal{F}^k]-\mathcal{V}^k &\leq -\alpha_k\left(1-\frac{\alpha_k}{4\beta_k}C_g^2L_f^2\right)\|\nabla F(\theta^k)\|^2+\frac{L}{2}C_g^2C_f^2\alpha_k^2 \\
&\quad +(1+\beta_k)\mathbb{E}\left[\|g(\theta^k)-y^{k+1}\|^2|\mathcal{F}^k\right]-\|g(\theta^{k-1})-y^k\|^2 \\
&\overset{(a)}{\leq} -\alpha_k\left(1-\frac{\alpha_k}{4\beta_k}C_g^2L_f^2\right)\|\nabla F(\theta^k)\|^2+\frac{L}{2}C_g^2C_f^2\alpha_k^2 \\
&\quad +2(1+\beta_k)\beta_k^2V_g^2+\big((1+\beta_k)(1-\beta_k)^2-1\big)\|g(\theta^{k-1})-y^k\|^2 \\
&\quad +4(1+\beta_k)(1-\beta_k)^2C_g^4C_f^2\alpha_k^2 \\
&\overset{(b)}{\leq} -\alpha_k\left(1-\frac{\alpha_k}{4\beta_k}C_g^2L_f^2\right)\|\nabla F(\theta^k)\|^2+\frac{L}{2}C_g^2C_f^2\alpha_k^2 \\
&\quad +2(1+\beta_k)\beta_k^2V_g^2+4C_g^4C_f^2\alpha_k^2
\end{aligned} \tag{2.40}$$

where (a) follows from Lemma 1, and (b) uses that $(1+\beta_k)(1-\beta_k)^2=(1-\beta_k^2)(1-\beta_k)\leq 1$ twice.

Select $\alpha_k=\frac{2\beta_k}{C_g^2L_f^2}$ so that $1-\frac{\alpha_k}{4\beta_k}C_g^2L_f^2=\frac{1}{2}$, and define (with $\beta_k\in(0,1)$)

$$B_1:=\frac{L}{2}C_g^2C_f^2+4V_g^2+4C_g^4C_f^2\geq\frac{L}{2}C_g^2C_f^2+2(1+\beta_k)V_g^2+4C_g^4C_f^2. \tag{2.41}$$

Taking expectation over $\mathcal{F}^k$ on both sides of (2.40), then it follows that

$$\mathbb{E}[\mathcal{V}^{k+1}]\leq\mathbb{E}[\mathcal{V}^k]-\frac{\alpha_k}{2}\mathbb{E}[\|\nabla F(\theta^k)\|^2]+B_1\alpha_k^2. \tag{2.42}$$

Rearranging terms, we have

$$\frac{\sum_{k=0}^K\alpha_k\mathbb{E}[\|\nabla F(\theta^k)\|^2]}{\sum_{k=0}^K\alpha_k}\leq\frac{2\mathcal{V}^0+2B_1\sum_{k=0}^K\alpha_k^2}{\sum_{k=0}^K\alpha_k}.$$

Choosing the stepsize as $\alpha_k=\frac{1}{\sqrt{K}}$ leads to

$$\frac{\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla F(\theta^k)\|^2]}{K}\leq\frac{2\mathcal{V}^0+2B_1}{\sqrt{K}}$$

from which the proof is complete.

2.6.2 Proof of Theorem 2

2.6.2.1 Supporting lemmas

We first present the essential lemmas that will lead to Theorem 2.

Lemma 3 *Under Assumption 5, the parameters $\{h^k, \hat{v}^k\}$ of Adam SCSC in Algorithm 2 satisfy*

$$\|h^k\| \leq C_g C_f, \quad \forall k; \quad \hat{v}_i^k \leq C_g^2 C_f^2, \quad \forall k, i. \tag{2.43}$$

Proof: Using Assumption 5, it follows that $\|\boldsymbol{\nabla}^k\| = \|\nabla g(\theta^k; \phi^k) \nabla f(y^{k+1}; \xi^k)\| \leq C_g C_f$. Therefore, from the update (2.16a), we have

$$\|h^{k+1}\| \leq \eta_1 \|h^k\| + (1-\eta_1)\|\boldsymbol{\nabla}^k\| \leq \eta_1 \|h^k\| + (1-\eta_1) C_g C_f.$$

Since $\|h^1\| \leq C_g C_f$, if follows by induction that $\|h^{k+1}\| \leq C_g C_f$.

Similarly, from the update (2.16b), we have

$$\begin{aligned} \hat{v}_i^{k+1} &\leq \max\{\hat{v}_i^k, \eta_2 \hat{v}_i^k + (1-\eta_2)(\nabla_i^k)^2\} \\ &\leq \max\{\hat{v}_i^k, \eta_2 \hat{v}_i^k + (1-\eta_2) C_g^2 C_f^2\}. \end{aligned}$$

Since $v_i^1 = \hat{v}_i^1 \leq C_g^2 C_f^2$, by induction, $\hat{v}_i^{k+1} \leq C_g^2 C_f^2$.

Lemma 4 *Under Assumption 5, the iterates $\{\theta^k\}$ of Adam SCSC in Algorithm 2 satisfy*

$$\left\|\theta^{k+1} - \theta^k\right\|^2 \leq \alpha_k^2 d (1-\eta_2)^{-1} (1-\gamma)^{-1} \tag{2.44}$$

where d is the dimension of θ, $\eta_1 < \sqrt{\eta_2} < 1$, and $\gamma := \eta_1^2/\eta_2$.

Proof: Choosing $\eta_1 < 1$ and defining $\gamma := \eta_1^2/\eta_2$, it can be verified that

$$\begin{aligned}
|h_i^{k+1}| &= \left|\eta_1 h_i^k + (1-\eta_1)\nabla_i^k\right| \leq \eta_1 |h_i^k| + |\nabla_i^k| \\
&\leq \eta_1 \left(\eta_1 |h_i^{k-1}| + |\nabla_i^{k-1}|\right) + |\nabla_i^k| \\
&\leq \sum_{l=0}^{k} \eta_1^{k-l} |\nabla_i^l| = \sum_{l=0}^{k} \sqrt{\gamma}^{k-l} \sqrt{\eta_2}^{k-l} |\nabla_i^l| \\
&\overset{(a)}{\leq} \left(\sum_{l=0}^{k} \gamma^{k-l}\right)^{\frac{1}{2}} \left(\sum_{l=0}^{k} \eta_2^{k-l} (\nabla_i^l)^2\right)^{\frac{1}{2}} \\
&\leq (1-\gamma)^{-\frac{1}{2}} \left(\sum_{l=0}^{k} \eta_2^{k-l} (\nabla_i^l)^2\right)^{\frac{1}{2}}
\end{aligned} \tag{2.45}$$

where (a) follows from the Cauchy-Schwartz inequality.

For $\hat{v}_i^k$, first we have that $\hat{v}_i^1 \geq (1-\eta_2)(\nabla_i^1)^2$. Then since

$$\hat{v}_i^{k+1} \geq \eta_2 \hat{v}_i^k + (1-\eta_2)(\nabla_i^k)^2$$

by induction we have

$$\hat{v}_i^{k+1} \geq (1-\eta_2) \sum_{l=0}^{k} \eta_2^{k-l} (\nabla_i^l)^2. \tag{2.46}$$

Using (2.45) and (2.46), we have

$$\begin{aligned}
|h_i^{k+1}|^2 \leq& (1-\gamma)^{-1} \left(\sum_{l=0}^{k} \eta_2^{k-l} (\nabla_i^l)^2\right) \\
\leq& (1-\eta_2)^{-1} (1-\gamma)^{-1} \hat{v}_i^{k+1}.
\end{aligned}$$

From the update (2.16c), we have

$$\begin{aligned}
\|\theta^{k+1} - \theta^k\|^2 &= \alpha_k^2 \sum_{i=1}^{d} \left(\epsilon + \hat{v}_i^{k+1}\right)^{-1} |h_i^{k+1}|^2 \\
&\leq \alpha_k^2 d (1-\eta_2)^{-1} (1-\gamma)^{-1}
\end{aligned} \tag{2.47}$$

which completes the proof.

2.6.2.2 Remaining steps towards Theorem 2

We are ready to prove Theorem 2. We re-write the Lyapunov function (2.22) as

$$\mathcal{V}^k := F(\theta^k) - F(\theta^*) - c_k \left\langle \nabla F(\theta^{k-1}), \frac{h^k}{\sqrt{\epsilon + \hat{v}^k}} \right\rangle + c \left\| g(\theta^{k-1}) - y^k \right\|^2 \tag{2.48}$$

where $\{c_k\}$ and c are constants to be determined later.

Using the smoothness of $F(\theta^k)$, we have

$$\begin{aligned} & F(\theta^{k+1}) - F(\theta^k) \\ & \leq \langle \nabla F(\theta^k), \theta^{k+1} - \theta^k \rangle + \frac{L}{2} \|\theta^{k+1} - \theta^k\|^2 \\ & = -\alpha_k \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} h^{k+1} \rangle + \frac{L}{2} \|\theta^{k+1} - \theta^k\|^2 \end{aligned} \tag{2.49}$$

where $\hat{V}^{k+1} := \mathrm{diag}(\hat{v}^{k+1})$ and $(\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}}$ is understood entry-wise.

Recalling $\boldsymbol{\nabla}^k := \nabla g(\theta^k; \phi^k) \nabla f(y^{k+1}; \xi^k)$, the inner product in (2.49) can be decomposed as

$$\begin{aligned} & - \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} h^{k+1} \rangle \qquad (2.50) \\ = & \underbrace{-(1 - \eta_1) \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} \boldsymbol{\nabla}^k \rangle}_{I_1^k} \\ & \underbrace{-\eta_1 \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} h^k \rangle}_{I_2^k} \\ & \underbrace{-\langle \nabla F(\theta^k), \left((\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} - (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} \right) h^{k+1} \rangle}_{I_3^k}. \end{aligned}$$

By defining $\bar{\boldsymbol{\nabla}}^k := \nabla g(\theta^k; \phi^k) \nabla f(g(\theta^k); \xi^k)$, we have

$$\begin{aligned} I_1^k = & - (1 - \eta_1) \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} \bar{\boldsymbol{\nabla}}^k \rangle \\ & - (1 - \eta_1) \langle \nabla F(\theta^k), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} \big(\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \big) \rangle. \end{aligned} \tag{2.51}$$

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k and ξ^k on I_1^k, we have

$$\begin{aligned} & \mathbb{E}\left[I_1^k | \mathcal{F}^k \right] \overset{(a)}{\leq} -(1 - \eta_1) \left\| \nabla F(\theta^k) \right\|^2_{(\epsilon I + \hat{V}^k)^{-\frac{1}{2}}} \\ & + (1 - \eta_1) \big\| (\epsilon I + \hat{V}^k)^{-\frac{1}{4}} \nabla F(\theta^k) \big\| \mathbb{E}\Big[\big\| (\epsilon I + \hat{V}^k)^{-\frac{1}{4}} \big(\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \big) \big\| \big| \mathcal{F}^k \Big] \end{aligned} \tag{2.52}$$

where (a) uses $\mathbb{E}\left[\bar{\boldsymbol{\nabla}}^k|\mathcal{F}^k\right] = \nabla F(\theta^k)$.

Expanding the second term in the RHS of (2.52), we have

$$\begin{aligned}
\mathbb{E}\left[I_1^k|\mathcal{F}^k\right] \overset{(b)}{\leq} & -(1-\eta_1)\left(1-\frac{\alpha_k}{4\beta_k}\right)\left\|\nabla F(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^k)^{-\frac{1}{2}}} \\
& + \frac{\beta_k}{\alpha_k}\mathbb{E}\left[\left\|\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k\right\|^2_{(\epsilon I+\hat{V}^k)^{-\frac{1}{2}}}|\mathcal{F}^k\right] \\
\overset{(c)}{\leq} & -(1-\eta_1)\left(1-\frac{\alpha_k}{4\beta_k}\right)\left\|\nabla F(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^k)^{-\frac{1}{2}}} \\
& + \frac{\beta_k}{\alpha_k}\epsilon^{-\frac{1}{2}}C_g^2L_f^2\mathbb{E}\left[\left\|g(\theta^k) - y^{k+1}\right\|^2|\mathcal{F}^k\right] \\
\overset{(d)}{\leq} & -(1-\eta_1)\left(1-\frac{\alpha_k}{4\beta_k}\right)(\epsilon I + C_g^2C_f^2)^{-\frac{1}{2}}\left\|\nabla F(\theta^k)\right\|^2 \\
& + \frac{\beta_k}{\alpha_k}\epsilon^{-\frac{1}{2}}C_g^2L_f^2\mathbb{E}\left[\left\|g(\theta^k) - y^{k+1}\right\|^2|\mathcal{F}^k\right]
\end{aligned} \tag{2.53}$$

where (b) is due to the Young's inequality $ab \leq \frac{a^2}{4\beta_k} + \beta_k b^2$ and $1-\eta_1 \leq 1$; (c) follows from Assumptions 1 and 2; and, (d) uses Lemma 3.

Likewise, for I_2^k, we have

$$\begin{aligned}
\mathbb{E}\left[I_2^k|\mathcal{F}^k\right] = & -\eta_1\langle\nabla F(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\rangle \\
& -\eta_1\langle\nabla F(\theta^k) - \nabla F(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\rangle \\
\overset{(a)}{\leq} & -\eta_1\langle\nabla F(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\rangle + \eta_1 L\alpha_{k-1}^{-1}\|\theta^k - \theta^{k-1}\|^2 \\
\overset{(b)}{\leq} & -\eta_1\langle\nabla F(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\rangle + \alpha_{k-1}\eta_1 Ld(1-\eta_2)^{-1}(1-\gamma)^{-1} \\
\overset{(c)}{=} & -\eta_1(I_1^{k-1} + I_2^{k-1} + I_3^{k-1}) + \alpha_{k-1}\eta_1 Ld(1-\eta_2)^{-1}(1-\gamma)^{-1}
\end{aligned} \tag{2.54}$$

where (a) follows from the L-smoothness of $F(\theta)$ implied by Assumptions 1 and 2; (b) follows from Lemma 4; and (c) uses again the decomposition (2.50).

Use $h_i^k, v_i^k, \theta_i^k, \nabla_i^k$ to denote the ith entry of $h^k, v^k, \theta^k, \boldsymbol{\nabla}^k$. We have $|\nabla_i F(\boldsymbol{\theta}^k)| \leq \|\nabla F(\boldsymbol{\theta}^k)\|$, $|h_i^{k+1}| \leq \|\mathbf{h}^{k+1}\|$ and $(\epsilon + \hat{v}_i^k)^{\frac{1}{2}} \geq (\epsilon + \hat{v}_i^{k+1})^{\frac{1}{2}}$ as $\hat{v}_i^{k+1} = \max\{\cdot, \hat{v}_i^k\} \geq \hat{v}_i^k$.

For I_3^k, we have

$$\begin{aligned}\mathbb{E}\left[I_3^k|\mathcal{F}^k\right] =& -\sum_{i=1}^{d}\nabla_i F(\theta^k)\left((\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k})^{-\frac{1}{2}}\right)h_i^{k+1}\\ \leq&\|\nabla F(\theta^k)\|\|h^{k+1}\|\sum_{i=1}^{d}\left((\epsilon+\hat{v}_i^{k})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\\ \overset{(d)}{\leq}&C_g^2C_f^2\sum_{i=1}^{d}\left((\epsilon+\hat{v}_i^{k})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\end{aligned}\tag{2.55}$$

where (d) follows from Assumption 5 and Lemma 3.

Recalling the definition of $\mathcal{V}^k$ in (2.22), we have

$$\begin{aligned}&\mathcal{V}^{k+1}-\mathcal{V}^k\\ =&F(\theta^{k+1})-F(\theta^k)-c_{k+1}\left\langle\nabla F(\theta^k),(\epsilon I+\hat{V}^{k+1})^{-\frac{1}{2}}h^{k+1}\right\rangle\\ &+c\|g(\theta^k)-y^{k+1}\|^2+c_k\left\langle\nabla F(\theta^{k-1}),(\epsilon I+\hat{V}^{k})^{-\frac{1}{2}}h^{k}\right\rangle\\ &-c\|g(\theta^{k-1})-y^k\|^2\\ \overset{(2.49)}{\leq}&-(\alpha_k+c_{k+1})\langle\nabla F(\theta^k),(\epsilon I+\hat{V}^{k+1})^{-\frac{1}{2}}h^{k+1}\rangle\\ &+\frac{L}{2}\|\theta^{k+1}-\theta^k\|^2+c\|g(\theta^k)-y^{k+1}\|^2\\ &+c_k\left\langle\nabla F(\theta^{k-1}),(\epsilon I+\hat{V}^{k})^{-\frac{1}{2}}h^{k}\right\rangle-c\|g(\theta^{k-1})-y^k\|^2.\end{aligned}\tag{2.56}$$

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k and ξ^k on both sides of (2.56), we have

$$\begin{aligned}
&\mathbb{E}[\mathcal{V}^{k+1}|\mathcal{F}^k]-\mathcal{V}^k\\
\leq&(\alpha_k+c_{k+1})\mathbb{E}[I_1^k+I_2^k+I_3^k \mid \mathcal{F}^k]+\frac{L}{2}\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2 \mid \mathcal{F}^k]\\
&+c_k(I_1^{k-1}+I_2^{k-1}+I_3^{k-1})+c\mathbb{E}[\|g(\theta^k)-y^{k+1}\|^2 \mid \mathcal{F}^k]\\
&-c\|g(\theta^{k-1})-y^k\|^2\\
\overset{(e)}{\leq}&-(\alpha_k+c_{k+1})(1-\eta_1)\left(1-\frac{\alpha_k}{4\beta_k}\right)(\varepsilon+C_g^2C_f^2)^{-\frac{1}{2}}\|\nabla F(\theta^k)\|^2\\
&-((\alpha_k+c_{k+1})\eta_1-c_k)\left(I_1^{k-1}+I_2^{k-1}+I_3^{k-1}\right)\\
&+(\alpha_k+c_{k+1})\alpha_{k-1}\eta_1 Ld(1-\eta_2)^{-1}(1-\gamma)^{-1}\\
&+(\alpha_k+c_{k+1})C_g^2C_f^2\sum_{i=1}^d\left((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\\
&+\left(c+\frac{\alpha_k+c_{k+1}}{\alpha_k}\beta_k\epsilon^{-\frac{1}{2}}C_g^2L_f^2\right)\mathbb{E}[\|g(\theta^k)-y^{k+1}\|^2 \mid \mathcal{F}^k]\\
&+\frac{L}{2}\alpha_k^2(1-\eta_2)^{-1}(1-\gamma)^{-1}-c\|g(\theta^{k-1})-y^k\|^2
\end{aligned} \tag{2.57}$$

where (e) substitutes $\mathbb{E}[I_1^k+I_2^k+I_3^k \mid \mathcal{F}^k]$ by (2.53)-(2.55) and applies Lemma 4.

Selecting $\alpha_{k+1}\leq\alpha_k$ and $c_k:=\sum_{p=k}^{\infty}\prod_{j=k}^{p}\eta_1\alpha_p\leq(1-\eta_1)^{-1}\alpha_k$, we have

$$\begin{aligned}
\frac{\alpha_k+c_{k+1}}{\alpha_k}\beta_k\epsilon^{-\frac{1}{2}}C_g^2L_f^2 &\leq \frac{\alpha_k+(1-\eta_1)^{-1}\alpha_{k+1}}{\alpha_k}\beta_k\epsilon^{-\frac{1}{2}}C_g^2L_f^2\\
&\leq \frac{\alpha_k+(1-\eta_1)^{-1}\alpha_k}{\alpha_k}\beta_k\epsilon^{-\frac{1}{2}}C_g^2L_f^2\\
&:= c\beta_k
\end{aligned}$$

where we define $c:=(1+(1-\eta_1)^{-1})\epsilon^{-\frac{1}{2}}C_g^2L_f^2$.

Therefore, applying Lemma 1, we have

$$
\begin{aligned}
&\Big(c+\frac{\alpha_k+c_{k+1}}{\alpha_k}\beta_k\epsilon^{-\frac{1}{2}}C_g^2L_f^2\Big)\mathbb{E}[\|g(\theta^k)-y^{k+1}\|^2 \mid \mathcal{F}^k] \\
&\quad - c\|g(\theta^{k-1})-y^k\|^2 \\
\leq & c(1+\beta_k)\mathbb{E}[\|g(\theta^k)-y^{k+1}\|^2 \mid \mathcal{F}^k] - c\|g(\theta^{k-1})-y^k\|^2 \\
\leq & c\left((1+\beta_k)(1-\beta_k)^2-1\right)\|g(\theta^{k-1})-y^k\|^2 \\
&\quad + 4c(1+\beta_k)(1-\beta_k)^2C_g^2\|\theta^k-\theta^{k-1}\|^2 + 2c(1+\beta_k)\beta_k^2V_g^2 \\
\leq & 4c(1+\beta_k)(1-\beta_k)^2C_g^2\left(\alpha_{k-1}^2 d(1-\eta_2)^{-1}(1-\gamma)^{-1}\right) \\
&\quad + 2c(1+\beta_k)\beta_k^2V_g^2 \\
\overset{(f)}{\leq} & 4cC_g^2\alpha_{k-1}^2d(1-\eta_2)^{-1}(1-\gamma)^{-1} + 2c(1+\beta_k)\beta_k^2V_g^2
\end{aligned}
\tag{2.58}
$$

where (f) follows from $(1+\beta_k)(1-\beta_k)^2 = (1-\beta_k^2)(1-\beta_k) \leq 1$.

Selecting $c_k := \sum_{p=k}^{\infty}\prod_{j=k}^{p}\eta_1\alpha_p$ implies $(\alpha_k+c_{k+1})\eta_1 = c_k$. We thus obtain from (2.57) and (2.58) that

$$
\begin{aligned}
&\mathbb{E}[\mathcal{V}^{k+1}|\mathcal{F}^k]-\mathcal{V}^k \\
\leq & -(\alpha_k+c_{k+1})(1-\eta_1)\Big(1-\frac{\alpha_k}{4\beta_k}\Big)(\varepsilon+C_g^2C_f^2)^{-\frac{1}{2}}\|\nabla F(\theta^k)\|^2 \\
&\quad + 4cC_g^2\alpha_{k-1}^2d(1-\eta_2)^{-1}(1-\gamma)^{-1} + 2c\beta_k^2(1+\beta_k)V_g^2 \\
&\quad + (1-\eta_1)^{-1}Ld(1-\eta_2)^{-1}(1-\gamma)^{-1}\alpha_k\alpha_{k-1} \\
&\quad + (\alpha_k+c_{k+1})C_g^2C_f^2\sum_{i=1}^{d}\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big) \\
&\quad + \frac{L}{2}\alpha_k^2(1-\eta_2)^{-1}(1-\gamma)^{-1}.
\end{aligned}
\tag{2.59}
$$

Defining $\tilde{\eta} := (1-\eta_1)^{-1}(1-\eta_2)^{-1}(1-\gamma)^{-1}$

and rearranging terms in (2.59) and telescoping from $k = 0, \cdots, K-1$, we have

$$
\begin{aligned}
&\sum_{k=0}^{K-1} \alpha_k (1-\eta_1) \left(1 - \frac{\alpha_k}{4\beta_k}\right) (\varepsilon + C_g^2 C_f^2)^{-\frac{1}{2}} \mathbb{E}[\|\nabla F(\theta^k)\|^2] \\
\le& \mathcal{V}^0 - \mathbb{E}[\mathcal{V}^K] + \sum_{k=0}^{K-1} \left(4cC_g^2 \alpha_{k-1}^2 (1-\eta_1)\tilde{\eta} + 2c\beta_k^2(1+\beta_k)V_g^2\right) + \sum_{k=0}^{K-1} \left(\tilde{\eta} L d \alpha_{k-1}^2 + \frac{L}{2}(1-\eta_1)\tilde{\eta}\alpha_k^2\right) \\
&+ \sum_{k=0}^{K-1} (\alpha_k + c_{k+1}) C_g^2 C_f^2 \sum_{i=1}^{d} \left((\epsilon + \hat{v}_i^k)^{-\frac{1}{2}} - (\epsilon + \hat{v}_i^{k+1})^{-\frac{1}{2}}\right) \\
\overset{(g)}{\le}& \mathcal{V}^0 + (1-\eta_1)^{-1} \alpha_k C_g C_f d (1-\eta_1)\tilde{\eta} \\
&+ \sum_{k=0}^{K-1} \left(4cC_g^2 \alpha_{k-1}^2 (1-\eta_1)\tilde{\eta} + 2c\beta_k^2(1+\beta_k)V_g^2\right) + \sum_{k=0}^{K-1} \left(\tilde{\eta} L d \alpha_{k-1}^2 + \frac{L}{2}(1-\eta_1)\tilde{\eta}\alpha_k^2\right) \\
&+ (1 + (1-\eta_1)^{-1}) \alpha_0 C_g^2 C_f^2 \sum_{i=1}^{d} \left((\epsilon + \hat{v}_i^0)^{-\frac{1}{2}} - (\epsilon + \hat{v}_i^K)^{-\frac{1}{2}}\right)
\end{aligned}
$$

where (g) follows from $\alpha_k + c_{k+1} \le (1 + (1-\eta_1)^{-1})\alpha_k \le \alpha_0$ and the definition of $\mathcal{V}^k$ that

$$
\mathbb{E}[\mathcal{V}^k] \ge F(\theta^k) - F(\theta^*) + c\|g(\theta^{k-1}) - y^k\|^2 - (1-\eta_1)^{-1}\alpha_k C_g C_f d (1-\eta_2)^{-1}(1-\gamma)^{-1}.
$$

Select $\alpha_k = 2\beta_k = \alpha = \frac{1}{\sqrt{K}}$ so that $1 - \frac{\alpha_k}{4\beta_k} = \frac{1}{2}$. We have that

$$
\begin{aligned}
&\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}[\|\nabla F(\theta^k)\|^2] \\
\le& \frac{\mathcal{V}^0 + \sum_{k=0}^{K-1} \left(4C_g^2(1-\eta_1)\tilde{\eta} + V_g^2\right) c\alpha^2}{K\frac{\alpha(1-\eta_1)}{2}(\epsilon + C_g^2 C_f^2)^{-\frac{1}{2}}} + \frac{\sum_{k=0}^{K-1} \left(\tilde{\eta} L d + \frac{L}{2}(1-\eta_1)\tilde{\eta}\right)\alpha^2 + C_g C_f d \tilde{\eta}\alpha}{K\frac{\alpha(1-\eta_1)}{2}(\epsilon + C_g^2 C_f^2)^{-\frac{1}{2}}} \\
&+ \frac{(1 + (1-\eta_1)^{-1})\alpha_0 C_g^2 C_f^2 \sum_{i=1}^{d} (\epsilon + \hat{v}_i^0)^{-\frac{1}{2}}}{K\frac{\alpha(1-\eta_1)}{2}(\epsilon + C_g^2 C_f^2)^{-\frac{1}{2}}} \\
=& \frac{2(\epsilon + C_g^2 C_f^2)^{\frac{1}{2}}}{(1-\eta_1)} \left(\frac{\mathcal{V}^0 + (4C_g^2(1-\eta_1)\tilde{\eta} + V_g^2)c + (d + \frac{1}{2}(1-\eta_1))\tilde{\eta} L}{\sqrt{K}}\right. \\
&\left. + \frac{C_g C_f d \tilde{\eta} + (1 + (1-\eta_1)^{-1}) C_g^2 C_f^2 d \epsilon^{-\frac{1}{2}}}{K}\right) \\
\le& \frac{2(\epsilon + C_g^2 C_f^2)^{\frac{1}{2}}}{(1-\eta_1)} \left(\frac{\mathcal{V}^0 + \left(4C_g^2\tilde{\eta} + V_g^2\right)c + 2d\tilde{\eta} L}{\sqrt{K}} + \frac{C_g C_f d \tilde{\eta}}{K} + \frac{(1 + (1-\eta_1)^{-1}) C_g^2 C_f^2 d \epsilon^{-\frac{1}{2}}}{K}\right)
\end{aligned}
$$

from which the proof is complete.

2.7 Convergence results of the multi-level SCSC

In this section, we establish the convergence results of the multi-level SCSC, and present the corresponding analysis.

2.7.1 Supporting lemma

We first prove a multi-level version of the tracking variance lemma.

Lemma 5 (Tracking variance of multi-level SCSC) *If Assumptions 1-4 hold, and y_n^{k+1} is generated by running the multi-level SCSC iteration* (2.18) *given θ^k, then the variance of y_n^{k+1} satisfies*

$$\begin{aligned}\mathbb{E}\left[\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\|^2 \mid \mathcal{F}^k\right] \leq & (1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 \\ & + 4(1-\beta_k)^2 C_n^2 \mathbb{E}\left[\|y_{n-1}^k - y_{n-1}^{k+1}\|^2 | \mathcal{F}^k\right] + 2\beta_k^2 V^2. \qquad (2.60)\end{aligned}$$

Proof: Use $\mathcal{F}^{k,n}$ to denote the σ-algebra generated by $\{\cdots, \theta^k, y_1^k, \ldots, y_{n-1}^k\}$ From the update (2.18), we have that

$$\begin{aligned}y_n^{k+1} - f_n(y_{n-1}^{k+1}) &= (1-\beta_k)\left(y_n^k - f_n(y_{n-1}^k)\right) + (1-\beta_k)\left(f_n(y_{n-1}^k) - f_n(y_{n-1}^{k+1})\right) \\ &\quad + \beta_k\left(f(y_{n-1}^{k+1};\xi_n^k) - f_n(y_{n-1}^{k+1})\right) + (1-\beta_k)\left(f(y_{n-1}^{k+1};\xi_n^k) - f(y_{n-1}^k;\xi_n^k)\right) \\ &= (1-\beta_k)(y_n^k - f_n(y_{n-1}^k)) + (1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3 \qquad (2.61)\end{aligned}$$

where we define the three terms as

$$\begin{aligned}T_1 &:= f_n(y_{n-1}^k) - f_n(y_{n-1}^{k+1})) \\ T_2 &:= f_n(y_{n-1}^{k+1};\xi_n^k) - f_n(y_{n-1}^{k+1}) \\ T_3 &:= f_n(y_{n-1}^{k+1};\xi_n^k) - f_n(y_{n-1}^k;\xi_n^k).\end{aligned}$$

Conditioned on $\mathcal{F}^k$, taking expectation over ϕ^k, we have

$$\mathbb{E}\left[(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3 | \mathcal{F}^k\right] = \mathbf{0}. \qquad (2.62)$$

Conditioned on $\mathcal{F}^{k,n} := \{\mathcal{F}^k, y_1^{k+1}, \ldots, y_{n-1}^{k+1}\}$, taking expectation on (2.61), we have

$$
\begin{aligned}
&\mathbb{E}[\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\|^2 \mid \mathcal{F}^{k,n}] \\
=&\mathbb{E}[\|(1-\beta_k)(y_n^k - f_n(y_{n-1}^k))\|^2|\mathcal{F}^k] + \mathbb{E}\left[\|(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3\|^2 \mid \mathcal{F}^{k,n}\right] \\
&+ 2\mathbb{E}\left[\left\langle(1-\beta_k)(y_n^k - f_n(y_{n-1}^k)), (1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3\right\rangle \mid \mathcal{F}^{k,n}\right] \\
=&(1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 + \mathbb{E}\left[\|(1-\beta_k)T_1 + \beta_k T_2 + (1-\beta_k)T_3\|^2 \mid \mathcal{F}^{k,n}\right] \\
\leq&(1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 + 2\mathbb{E}\left[\|(1-\beta_k)T_1 + \beta_k T_2\|^2 \mid \mathcal{F}^{k,n}\right] + 2(1-\beta_k)^2\mathbb{E}\left[\|T_3\|^2 \mid \mathcal{F}^{k,n}\right] \\
\leq&(1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 + 2(1-\beta_k)^2\mathbb{E}[\|T_1\|^2 \mid \mathcal{F}^{k,n}] + 2\beta_k^2\mathbb{E}[\|T_2\|^2 \mid \mathcal{F}^{k,n}] \\
&+ 2\beta_k(1-\beta_k)\left\langle T_1, \mathbb{E}[T_2 \mid \mathcal{F}^{k,n}]\right\rangle + 2(1-\beta_k)^2\mathbb{E}[\|T_3\|^2 \mid \mathcal{F}^{k,n}] \\
\leq&(1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 + 2(1-\beta_k)^2\mathbb{E}\left[\|f_n(y_{n-1}^k) - f_n(y_{n-1}^{k+1})\|^2|\mathcal{F}^k\right] \\
&+ 2(1-\beta_k)^2\mathbb{E}\left[\|f_n(y_{n-1}^k;\xi_n^k) - f_n(y_{n-1}^{k+1};\xi_n^k)\|^2|\mathcal{F}^k\right] + 2\beta_k^2V^2 \\
\leq&(1-\beta_k)^2\|y_n^k - f_n(y_{n-1}^k)\|^2 + 4(1-\beta_k)^2C_n^2\mathbb{E}\left[\|y_{n-1}^k - y_{n-1}^{k+1}\|^2|\mathcal{F}^k\right] + 2\beta_k^2V^2
\end{aligned}
$$

from which the proof is complete.

Define $f^{(n)}(\theta) := f_n \circ f_{n-1} \circ \cdots \circ f_1(\theta)$ and the stochastic compositional gradients as

$$
\begin{aligned}
\boldsymbol{\nabla}^k &:= \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(y_{N-2}^{k+1};\xi_{N-1}^k)\nabla f_N(y_{N-1}^{k+1};\xi_N^k) \\
\bar{\boldsymbol{\nabla}}^k &:= \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k).
\end{aligned}
$$

Thus, taking expectation with respect to $\xi_1^k, \ldots, \xi_N^k$, we have

$$
\begin{aligned}
\mathbb{E}\left[\boldsymbol{\nabla}^k \mid \mathcal{F}^{k,N}\right] - \bar{\boldsymbol{\nabla}}^k =& \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(y_{N-2}^{k+1};\xi_{N-1}^k)\nabla f_N(y_{N-1}^{k+1};\xi_N^k) \\
&- \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(y_{N-2}^{k+1};\xi_{N-1}^k)\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k) \\
&+ \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(y_{N-2}^{k+1};\xi_{N-1}^k)\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k) \\
&- \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k) \\
&\cdots \\
&+ \nabla f_1(\theta^k;\xi_1^k)\nabla f_2(y_1^{k+1};\xi_2^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k) \\
&- \nabla f_1(\theta^k;\xi_1^k)\nabla f_2(f_1(\theta^k);\xi_2^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k). \qquad (2.63)
\end{aligned}
$$

Since the nth difference term in (2.63) can be bounded by (for convenience, define $y_0^{k+1} = \theta^k$)

$$
\begin{aligned}
&\Big\|\mathbb{E}\Big[\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_n(y_{n-1}^{k+1};\xi_n^k)\cdots f_N(f^{(N-1)}(\theta^k);\xi_N^k)\\
&\quad -\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_n(f^{(n-1)}(\theta^k);\xi_n^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k)\mid\mathcal{F}^k\Big]\Big\|\\
\overset{(a)}{\leq}\;& \underbrace{\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{n-1}(y_{n-2}^{k+1};\xi_{n-1}^k)\nabla f_{n+1}(f^{(n)}(\theta^k);\xi_{n+1}^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k)\big\|^2\mid\mathcal{F}^k\Big]^{\frac{1}{2}}}_{I_n^k}\\
&\times\underbrace{\mathbb{E}\Big[\big\|\nabla f_n(y_{n-1}^{k+1};\xi_n^k)-\nabla f_n(f^{(n-1)}(\theta^k);\xi_n^k)\big\|^2\mid\mathcal{F}^k\Big]^{\frac{1}{2}}}_{J_n^k}
\end{aligned}
\tag{2.64}
$$

where (a) uses the Cauchy-Schwartz inequality.

For I_n^k, using Assumption m2, we have

$$
\begin{aligned}
I_n^k &= \mathbb{E}\Big[\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k)\big\|^2\mid\mathcal{F}^{k,N}\Big]\mid\mathcal{F}^k\Big]^{\frac{1}{2}}\\
&\leq \mathbb{E}\Bigg[\mathbb{E}\Big[\big\|\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k)\big\|^2\Big]\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\big\|^2\mid\mathcal{F}^{k,N}\Big]\mid\mathcal{F}^k\Bigg]^{\frac{1}{2}}\\
&\leq C_N\mathbb{E}\Bigg[\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\big\|^2\mid\mathcal{F}^{k,N}\Big]\mid\mathcal{F}^k\Bigg]^{\frac{1}{2}}\\
&\leq C_N\mathbb{E}\Bigg[\mathbb{E}\Big[\big\|\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\big\|^2\Big]\\
&\qquad\qquad\times\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-2}(f^{(N-3)}(\theta^k);\xi_{N-2}^k)\big\|^2\mid\mathcal{F}^k,y_1^{k+1},\ldots,y_{N-2}^{k+1}\Big]\mid\mathcal{F}^k\Bigg]^{\frac{1}{2}}\\
&\leq C_{N-1}C_N\mathbb{E}\Big[\mathbb{E}\Big[\big\|\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_{N-1}(f^{(N-2)}(\theta^k);\xi_{N-1}^k)\big\|^2\mid\mathcal{F}^k,y_1^{k+1},\ldots,y_{N-2}^{k+1}\Big]\mid\mathcal{F}^k\Big]^{\frac{1}{2}}\\
&\leq C_1\cdots C_{n-1}C_{n+1}\cdots C_N.
\end{aligned}
$$

For J_n^k, using Assumption m1, we have

$$
\begin{aligned}
J_n^k &= \mathbb{E}\Big[\big\|\nabla f_n(y_{n-1}^{k+1};\xi_n^k)-\nabla f_n(f^{(n-1)}(\theta^k);\xi_n^k)\big\|^2\mid\mathcal{F}^k\Big]^{\frac{1}{2}}\\
&\leq L_n\mathbb{E}\Big[\big\|y_{n-1}^{k+1}-f^{(n-1)}(y_{n-2}^{k+1})\big\|\mid\mathcal{F}^k\Big].
\end{aligned}
$$

Plugging the above two upper bounds into (2.64), we have

$$\begin{aligned}
\left\|\mathbb{E}\left[\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \mid \mathcal{F}^k\right]\right\| = &\Big\|\mathbb{E}\Big[\nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_n(y_{n-1}^{k+1};\xi_n^k)\cdots f_N(f^{(N-1)}(\theta^k);\xi_N^k) \\
&- \nabla f_1(\theta^k;\xi_1^k)\cdots\nabla f_n(f^{(n-1)}(\theta^k);\xi_n^k)\cdots\nabla f_N(f^{(N-1)}(\theta^k);\xi_N^k) \mid \mathcal{F}^k\Big]\Big\| \\
\le & C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \mathbb{E}\Big[\left\|y_{n-1}^{k+1} - f^{(n-1)}(\theta^k)\right\| \mid \mathcal{F}^k\Big] \\
\overset{(b)}{\le} & C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \mathbb{E}\Big[\left\|y_{n-1}^{k+1} - f_{n-1}(y_{n-2}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&+ C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \mathbb{E}\Big[\left\|f_{n-1}(y_{n-2}^{k+1}) - f^{(n-1)}(\theta^k)\right\| \mid \mathcal{F}^k\Big]
\end{aligned} \tag{2.65}$$

where (b) uses the triangular inequality.

Using the L_{n-1} Lipschitz continuity of $f^{(n-1)}$, we have

$$\begin{aligned}
\left\|\mathbb{E}\left[\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \mid \mathcal{F}^k\right]\right\| \le & C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \mathbb{E}\Big[\left\|y_{n-1}^{k+1} - f_{n-1}(y_{n-2}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&+ C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n L_{n-1} \mathbb{E}\Big[\left\|y_{n-2}^{k+1} - f^{(n-2)}(\theta^k)\right\| \mid \mathcal{F}^k\Big].
\end{aligned} \tag{2.66}$$

Repeating the steps in (2.65) and (2.66), we can recursively obtain

$$\begin{aligned}
\left\|\mathbb{E}\left[\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \mid \mathcal{F}^k\right]\right\| \le & C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \mathbb{E}\Big[\left\|y_{n-1}^{k+1} - f_{n-1}(y_{n-2}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&+ C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n L_{n-1} \mathbb{E}\Big[\left\|y_{n-2}^{k+1} - f_{n-2}(y_{n-3}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&+ C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \cdots L_{n-2} \mathbb{E}\Big[\left\|y_{n-3}^{k+1} - f_{n-3}(y_{n-4}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&+ \cdots + C_1\cdots C_{n-1}C_{n+1}\cdots C_N L_n \cdots L_2 \mathbb{E}\Big[\left\|y_1^{k+1} - f_1(\theta^k)\right\| \mid \mathcal{F}^k\Big] \\
\overset{(c)}{=} & \sum_{m=1}^{n-1} A_{m,n} \mathbb{E}\Big[\left\|y_m^{k+1} - f_m(y_{m-1}^{k+1})\right\| \mid \mathcal{F}^k\Big]
\end{aligned} \tag{2.67}$$

where (c) follows by defining

$$A_{m,n} := C_N \cdots C_{n+1} C_{n-1} \cdots C_1 L_n \cdots L_{m+1}. \tag{2.68}$$

Therefore, using Assumption m3, we have

$$\begin{aligned}
\left\|\mathbb{E}\left[\boldsymbol{\nabla}^k \mid \mathcal{F}^k\right] - \nabla F(\theta^k)\right\| &= \left\|\mathbb{E}\left[\boldsymbol{\nabla}^k \mid \mathcal{F}^k\right] - \mathbb{E}\left[\bar{\boldsymbol{\nabla}}^k \mid \mathcal{F}^k\right]\right\| \\
&= \left\|\mathbb{E}\left[\boldsymbol{\nabla}^k - \bar{\boldsymbol{\nabla}}^k \mid \mathcal{F}^k\right]\right\| \\
&\overset{(d)}{=} \sum_{n=2}^{N}\sum_{m=1}^{n-1} A_{m,n}\mathbb{E}\Big[\left\|y_m^{k+1} - f_m(y_{m-1}^{k+1})\right\| \mid \mathcal{F}^k\Big] \\
&= \sum_{n=1}^{N-1} A_n\mathbb{E}\Big[\left\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\right\| \mid \mathcal{F}^k\Big] \qquad (2.69)
\end{aligned}$$

where (d) follows from (2.67) and $A_n := \sum_{m=n+1}^{N-1} A_{n,m}$.

2.7.2 Remaining steps towards Theorem 3

Using the smoothness of $F(\theta^k)$, we have

$$\begin{aligned}
F(\theta^{k+1}) &\leq F(\theta^k) + \langle \nabla F(\theta^k), \theta^{k+1} - \theta^k\rangle + \frac{L}{2}\|\theta^{k+1} - \theta^k\|^2 \\
&= F(\theta^k) - \alpha_k\langle \nabla F(\theta^k), \boldsymbol{\nabla}^k\rangle + \frac{L}{2}\|\theta^{k+1} - \theta^k\|^2 \\
&= F(\theta^k) - \alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}\|\theta^{k+1} - \theta^k\|^2 + \alpha_k\langle \nabla F(\theta^k), \nabla F(\theta^k) - \boldsymbol{\nabla}^k\rangle.
\end{aligned}$$

Conditioned on $\mathcal{F}^k$, taking expectation over $\xi_1, \ldots, \xi_N$, we have

$$\begin{aligned}
&\mathbb{E}\left[F(\theta^{k+1})|\mathcal{F}^k\right] \\
\leq& F(\theta^k) - \alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}\mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k\right] + \alpha_k\left\langle \nabla F(\theta^k), \mathbb{E}\left[\nabla F(\theta^k) - \boldsymbol{\nabla}^k|\mathcal{F}^k\right]\right\rangle \\
\overset{(b)}{\leq}& F(\theta^k) - \alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2 + \alpha_k\|\nabla F(\theta^k)\|\left\|\mathbb{E}\left[\boldsymbol{\nabla}^k \mid \mathcal{F}^k\right] - \nabla F(\theta^k)\right\| \\
\overset{(c)}{\leq}& F(\theta^k) - \alpha_k\|\nabla F(\theta^k)\|^2 + \frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2 + \alpha_k\sum_{n=1}^{N-1} A_n\|\nabla F(\theta^k)\|\mathbb{E}\left[\left\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\right\||\mathcal{F}^k\right] \\
\overset{(d)}{\leq}& F(\theta^k) - \alpha_k\left(1 - \frac{\alpha_k}{4\beta_k}\sum_{n=1}^{N-1} A_n^2\right)\|\nabla F(\theta^k)\|^2 + \beta_k\sum_{n=1}^{N-1}\mathbb{E}\left[\left\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\right\|^2|\mathcal{F}^k\right] + \frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2
\end{aligned}$$

where (b) uses the Cauchy-Schwartz; (c) follows from (2.69); and (d) uses the Young's inequality.

Then with the definition of Lyapunov function in (2.24), it follows that

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}|\mathcal{F}^k] \leq\, & \mathcal{V}^k - \alpha_k\left(1-\frac{\alpha_k}{4\beta_k}\sum_{n=1}^{N-1}A_n^2\right)\|\nabla\mathcal{L}(\theta^k)\|^2+\frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2 \\
& + (1+2\beta_k)\sum_{n=1}^{N-1}\mathbb{E}\left[\left\|y_n^{k+1}-f_n(y_{n-1}^{k+1})\right\|^2|\mathcal{F}^k\right]-\sum_{n=1}^{N-1}\mathbb{E}\left[\left\|y_n^{k}-f_n(y_{n-1}^{k})\right\|^2|\mathcal{F}^k\right] \\
& - \beta_k\sum_{n=1}^{N-1}\mathbb{E}\left[\left\|y_n^{k+1}-f_n(y_{n-1}^{k+1})\right\|^2|\mathcal{F}^k\right] \\
\overset{(e)}{\leq}\, & \mathcal{V}^k - \alpha_k\left(1-\frac{\alpha_k}{4\beta_k}\sum_{n=1}^{N-1}A_n^2\right)\|\nabla\mathcal{L}(\theta^k)\|^2+\frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2+2(1+2\beta_k)\beta_k^2V^2 \\
& + \big((1+2\beta_k)(1-\beta_k)^2-1\big)\sum_{n=1}^{N-1}\mathbb{E}\left[\left\|y_n^{k}-f_n(y_{n-1}^{k})\right\|^2|\mathcal{F}^k\right] \\
& + 4(1+2\beta_k)(1-\beta_k)^2C_1^2\mathbb{E}\left[\|\theta^k-\theta^{k-1}\|^2|\mathcal{F}^k\right] \\
& + \sum_{n=2}^{N-1}\left[4(1+2\beta_k)(1-\beta_k)^2C_n^2+\gamma_n\right]\mathbb{E}\left[\|y_{n-1}^{k+1}-y_{n-1}^k\|^2|\mathcal{F}^k\right] \\
& - \sum_{n=2}^{N-1}\gamma_n\mathbb{E}\left[\|y_{n-1}^{k+1}-y_{n-1}^k\|^2|\mathcal{F}^k\right]-\beta_k\sum_{n=1}^{N-1}\mathbb{E}\left[\|y_n^{k+1}-f_n(y_{n-1}^{k+1})\|^2|\mathcal{F}^k\right] \\
\overset{(f)}{\leq}\, & \mathcal{V}^k - \alpha_k\left(1-\frac{\alpha_k}{4\beta_k}\sum_{n=1}^{N-1}A_n^2\right)\|\nabla\mathcal{L}(\theta^k)\|^2+\frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2+2(1+2\beta_k)\beta_k^2V^2 \\
& + 4C_1^2\mathbb{E}\left[\|\theta^k-\theta^{k-1}\|^2\mid\mathcal{F}^k\right]+\sum_{n=2}^{N-1}(4C_n^2+\gamma_n)\mathbb{E}\left[\|y_{n-1}^{k+1}-y_{n-1}^k\|^2\mid\mathcal{F}^k\right] \\
& - \sum_{n=2}^{N-1}\gamma_n\mathbb{E}\left[\|y_{n-1}^{k+1}-y_{n-1}^k\|^2\mid\mathcal{F}^k\right]-\beta_k\sum_{n=1}^{N-1}\mathbb{E}\left[\|y_n^{k+1}-f_n(y_{n-1}^{k+1}\|^2\mid\mathcal{F}^k\right] \quad (2.70)
\end{aligned}$$

where (e) follows from Lemma 5; uses that $(4(1+2\beta_k)(1-\beta_k)^2C_1^2)\,\mathbb{E}\left[\|y_0^{k+1}-y_0^k\|^2|\mathcal{F}^k\right]\leq 4C_1^2\mathbb{E}\left[\|\theta^k-\theta^{k-1}\|^2|\mathcal{F}^k\right]$; and $\gamma_n>0$ is a fixed constant.

On the other hand, from the update (2.18), we have that

$$(1-\beta_k)\left(y_{n-1}^{k+1}-y_{n-1}^k\right)=\beta_k\left(f(y_{n-2}^{k+1};\xi_{n-1}^k)-y_{n-1}^{k+1}\right)+(1-\beta_k)\left(f(y_{n-2}^{k+1};\xi_{n-1}^k)-f(y_{n-2}^{k};\xi_{n-1}^k)\right).$$

Squaring both sides, and taking expectation conditioned on $\mathcal{F}^k$, we have

$$\begin{aligned}
&\mathbb{E}\left[\|y_{n-1}^{k+1} - y_{n-1}^{k}\|^2 \mid \mathcal{F}^k\right] \\
&\overset{(g)}{\leq} 2\left(\frac{\beta_k}{1-\beta_k}\right)^2 \mathbb{E}\left[\|f_{n-1}(y_{n-2}^{k+1};\xi_{n-1}^k) - f_{n-1}(y_{n-2}^{k+1}) + f_{n-1}(y_{n-2}^{k+1}) - y_{n-1}^{k+1}\|^2 \mid \mathcal{F}^k\right] \\
&\quad + 2\mathbb{E}\left[\left\|f_{n-1}(y_{n-2}^{k+1};\xi_{n-1}^k) - f_{n-1}(y_{n-2}^{k};\xi_{n-1}^k)\right\|^2 \mid \mathcal{F}^k\right] \\
&\leq 2\left(\frac{\beta_k}{1-\beta_k}\right)^2 \mathbb{E}\left[\|y_{n-1}^{k+1} - f_{n-1}(y_{n-2}^{k+1})\|^2 \mid \mathcal{F}^k\right] \\
&\quad + 2C_{n-1}^2\mathbb{E}\left[\|y_{n-2}^{k+1} - y_{n-2}^{k}\|^2 \mid \mathcal{F}^k\right] + 2\left(\frac{\beta_k}{1-\beta_k}\right)^2 V^2 \qquad (2.71)
\end{aligned}$$

where (g) follows from the Young's inequality.

Plugging (2.71) into (2.70), we have

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}|\mathcal{F}^k] \leq{}& \mathcal{V}^k - \alpha_k\left(1 - \frac{\alpha_k}{4\beta_k}\sum_{n=1}^{N-1} A_n^2\right)\|\nabla\mathcal{L}(\theta^k)\|^2 + \frac{L}{2}C_1^2\cdots C_N^2\alpha_k^2 + 4C_1^2\|\theta^k - \theta^{k-1}\|^2 \\
&+ \left(2(1+2\beta_k)\beta_k^2 + 2\left(\frac{\beta_k}{1-\beta_k}\right)^2\sum_{n=2}^{N-1}(4C_n^2+\gamma_n)\right)V^2 \\
&+ 2\left(\frac{\beta_k}{1-\beta_k}\right)^2\sum_{n=2}^{N-1}(4C_n^2+\gamma_n)\mathbb{E}\left[\|y_{n-1}^{k+1} - f_{n-1}(y_{n-2}^{k+1})\|^2 \mid \mathcal{F}^k\right] \\
&+ 2\sum_{n=2}^{N-1}(4C_n^2+\gamma_n)C_{n-1}^2\mathbb{E}\left[\|y_{n-2}^{k+1} - y_{n-2}^k\|^2|\mathcal{F}^k\right] \\
&- \sum_{n=2}^{N-1}\gamma_n\mathbb{E}\left[\|y_{n-1}^{k+1} - y_{n-1}^k\|^2|\mathcal{F}^k\right] - \beta_k\sum_{n=1}^{N-1}\mathbb{E}\left[\|y_n^{k+1} - f_n(y_{n-1}^{k+1})\|^2|\mathcal{F}^k\right]. \qquad (2.72)
\end{aligned}$$

Choose parameters $\{\gamma_n\}$ and $\{\beta_k\}$ such that

$$\begin{aligned}
&2(4C_n^2+\gamma_n)C_{n-1}^2 \leq \gamma_{n-1} \\
&2\left(\frac{\beta_k}{1-\beta_k}\right)^2(4C_n^2+\gamma_n) \leq \beta_k.
\end{aligned}$$

For γ_n, the condition can be satisfied by choosing

$$\gamma_{N-1} = 0,\ \gamma_{N-2} = 8C_{N-1}^2C_{N-2}^2,\ \gamma_{N-3} = 16C_{N-1}^2C_{N-2}^2C_{N-3}^2 + 8C_{N-2}^2C_{N-3}^2, \quad \cdots \qquad (2.73)$$

For β_k, the condition can be satisfied by solving following inequality that always has a solution

$$\beta_k \leq \frac{1 - 2\beta_k + (\beta_k)^2}{\gamma_{n-1} C_{n-1}^2}. \tag{2.74}$$

Select $\beta_k = \beta \leq \frac{1}{2}$ and $\alpha_k = \alpha = \frac{2\beta}{\sum_{n=1}^{N-1} A_n^2}$ so that $1 - \frac{\alpha_k}{4\beta_k} \sum_{n=1}^{N-1} A_n^2 = \frac{1}{2}$, and define

$$B_2 := \left(\frac{L}{2} + 4C_1^2 + 8C_1^2 C_2^2 + 2\gamma_2 C_1^2\right) C_1^2 \cdots C_N^2 \quad \text{and} \quad B_3 := 4\left(1 + 2\sum_{n=2}^{N-1}(4C_n^2 + \gamma_n)\right)V^2.$$

Plugging into (2.72) leads to

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}] \leq\, & \mathbb{E}[\mathcal{V}^k] - \frac{\alpha}{2}\mathbb{E}[\|\nabla \mathcal{L}(\theta^k)\|^2] + \frac{L}{2} C_1^2 \cdots C_N^2 \alpha^2 + 2(4C_2^2 + \gamma_2 + 2) C_1^2 \mathbb{E}\left[\|\theta^k - \theta^{k-1}\|^2\right] \\
& + \left(2(1+2\beta)\beta^2 + 2\left(\frac{\beta}{1-\beta}\right)^2 \sum_{n=2}^{N-1}(4C_n^2 + \gamma_n)\right)V^2 \\
\leq\, & \mathbb{E}[\mathcal{V}^k] - \frac{\alpha}{2}\mathbb{E}[\|\nabla \mathcal{L}(\theta^k)\|^2] + \left(\frac{L}{2} + 4C_1^2 + 8C_1^2 C_2^2 + 2\gamma_2\right) C_1^2 \cdots C_N^2 \alpha^2 \\
& + 2\left(1 + 2\beta + 4\sum_{n=2}^{N-1}\left[4C_n^2 + \gamma_n\right]\right)V^2\beta^2 \\
\leq\, & \mathbb{E}[\mathcal{V}^k] - \frac{\alpha}{2}\mathbb{E}[\|\nabla \mathcal{L}(\theta^k)\|^2] + B_2\alpha^2 + B_3\beta^2
\end{aligned} \tag{2.75}$$

Choosing the stepsize as $\alpha_k = \frac{c_\alpha}{\sqrt{K}}$ leads to

$$\frac{\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla F(\theta^k)\|^2]}{K} \leq \frac{2\mathcal{V}^0}{K\alpha} + 2B\alpha + 2B_3\frac{\beta^2}{\alpha} = \frac{2\mathcal{V}^0 + 2(B + B_3(\sum_{n=1}^{N-1} A_n^2)^2/4)}{\sqrt{K}}.$$

This completes the proof of Theorem 3.

CHAPTER 3

Single-loop Stochastic Algorithms for Stochastic Bilevel Optimization

3.1 Introduction

In this chapter, we consider solving the stochastic problems of the following form

$$\min_{\theta \in \mathbb{R}^d} \quad F(\theta) := \mathbb{E}_{\xi}\left[f\left(\theta, y^*(\theta); \xi\right)\right] \qquad \text{(upper)} \tag{3.1a}$$

$$\text{s. t.} \quad y^*(\theta) \in \underset{y \in \mathbb{R}^{d'}}{\arg\min}\ \mathbb{E}_{\phi}[g(\theta, y; \phi)] \qquad \text{(lower)} \tag{3.1b}$$

where f and g are differentiable functions; and, ξ and ϕ are random variables. The problem (3.1) is often referred to as the stochastic *bilevel* problem, where the upper-level optimization problem depends on the solution of the lower-level optimization over $y \in \mathbb{R}^{d'}$, denoted as $y^*(\theta)$, which depends on the value of upper-level variable $\theta \in \mathbb{R}^d$.

Bilevel optimization has a long history in operations research. It can be viewed as a generalization of the classic two-stage stochastic programming [101], in which the upper-level objective function depends on the optimal lower-level objective value rather than the lower-level solution. Earlier works have studied applications in portfolio management and game theory [104]; see a survey [20]. Recently, bilevel optimization has gained growing popularity in a number of machine learning applications such as meta-learning [88], reinforcement learning [56, 37], hyper-parameter optimization [28], continual learning [6], and image processing [58]. In some of these applications, when the lower-level problem admits a closed-form solution, bilevel optimization also reduces to the stochastic compositional optimization [117, 32, 11].

Unlike single-level stochastic problems, algorithms tailored for solving bilevel stochastic problems are much less explored. This is partially because solving this class of problems via traditional optimization techniques faces a number of challenges. A key difficulty due to the nested structure is that (stochastic) gradient, a basic element in continuous optimization machinery, is prohibitively expensive or even impossible to compute. As we will show later, since computing an unbiased stochastic gradient of $F(\theta)$ requires solving the lower-level problem once, running stochastic gradient descent (SGD) on the upper-level problem essentially results in a double-loop algorithm which uses an iterative algorithm to solve the lower-level problem thousands or even millions of times.

3.1.1 Prior art

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

Bilevel optimization. Many recent efforts have been made to solve the bilevel optimization problems. One successful approach is to reformulate the bilevel problem as a single-level problem by replacing the lower-level problem by its optimality conditions [14, 57]. Recently, gradient-based first-order methods for bilevel optimization have gained popularity, where the idea is to iteratively approximate the (stochastic) gradient of the upper-level problem either in forward or backward manner [95, 28, 99, 35]. While most of these works assume the unique solution of the lower-level problem, cases where this assumption does not hold have been tackled in the recent work [68]. All these algorithms have excellent empirical performance, but many of them either provide no theoretical guarantees or only focus on the asymptotic performance analysis.

The non-asymptotic analysis of bilevel optimization algorithms has been recently studied in some *pioneering* works, e.g., [33, 37, 44], just to name a few. In both [33, 44], bilevel stochastic optimization algorithms have been developed that run in a double-loop manner. To achieve an ϵ-stationary point, they only need the sample complexity $\mathcal{O}(\epsilon^{-2})$ that is comparable to the complexity of SGD for the single-level case. Recently, a single-loop two-timescale stochastic approximation algorithm has been developed in [37] for the bilevel problem (3.1). Due to the nature of two-timescale update, it incurs the sub-optimal sample complexity $\mathcal{O}(\epsilon^{-2.5})$. Therefore, the existing single-loop solvers for bilevel problems are significantly slower than those for problems without bilevel compositions, but otherwise share many structures and properties.

Stochastic compositional optimization. When the lower-level problem in (3.1b) admits a smooth closed-form solution, the bilevel problem (3.1) reduces to stochastic compositional optimization

$$\min_{\theta \in \mathbb{R}^d} \; F(\theta) := \mathbb{E}_{\xi}\left[f\left(\theta, \mathbb{E}_{\phi}[g(\theta;\phi)];\xi\right)\right]. \tag{3.2}$$

Popular approaches tackling this class of problems use two sequences of variables being updated in two different time scales [117, 118]. However, the complexity of [117] and [118] is worse than $\mathcal{O}(\epsilon^{-2})$ of SGD for the non-compositional case. Building upon recent variance-reduction techniques, variance-reduced methods have been developed to solve a special class of the stochastic compositional problem with the *finite-sum structure*, e.g., [62, 131], but they usually operate in a double-loop manner.

While most of existing algorithms rely on either two-timescale or double-loop updates, the single-timescale single-loop approaches have been recently developed in [32, 11], which achieve the sample complexity $\mathcal{O}(\epsilon^{-2})$. These encouraging recent results imply that solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. However, whether the stochastic optimization techniques used therein permeate to solving more challenging bilevel problems remains unknown.

3.1.2 Our contributions

To this end, this chapter aims to develop a *single-loop single-timescale* stochastic algorithm, which, for the class of smooth bilevel problems, can match the sample complexity of SGD for single-level stochastic optimization problems.

In the context of existing methods, our contributions can be summarized as follows.

1. We develop a new stochastic gradient estimator tailored for a certain class of stochastic bilevel problems, which is motivated by an ODE analysis for the corresponding continuous-time deterministic problems. Our new stochastic bilevel gradient estimator is flexible to combine with any existing stochastic optimization algorithms for the single-level problems, and solve this class of stochastic bilevel problems as sample-efficient as single-level problems.

2. When we combine this stochastic gradient estimator with SGD for the upper-level update, we term it as the Single-Timescale stochAstic BiLevEl optimization (**STABLE**) method. In the nonconvex case, to achieve ϵ-stationary point of (3.1), STABLE only requires $\mathcal{O}(\epsilon^{-2})$ samples in total. In the strongly convex case, to achieve ϵ-optimal solution of (3.1), STABLE only requires $\mathcal{O}(\epsilon^{-1})$ samples. To the best of our knowledge, STABLE is the *first* bilevel algorithm achieving the order of sample complexity as SGD for the classic stochastic single-level problems.

Trade-off and limitations. While our new bilevel optimization algorithm significantly improves the sample complexity of existing algorithms, it pays the price of additional computation per iteration. Specifically, in order to better estimate the stochastic bilevel gradient, a matrix inversion and an eigenvalue truncation are needed per iteration, which cost $\mathcal{O}(d^3)$ computation for a $d \times d$ matrix. In contrast, some of recent works [33, 37, 44] reduce matrix inversion to more efficient computations of matrix-vector products, which cost $\mathcal{O}(d^2)$ computation per iteration. Therefore, our algorithm is preferable in the regime where the sampling is more costly than computation or the dimension d is relatively small.

3.1.3 Applications

Next we describe two popular applications, all of which can be formulated as a bilevel problem.

Hyper-parameter optimization. Hyper-parameter optimization aims to find the optimal hyper-parameter $\theta \in \mathbb{R}^d$ (e.g., learning rate, regularization coefficient, neural network architecture), which is used in training a model $w \in \mathbb{R}^d$ on the training set, such that the learned model achieves the low risk on the validation set. Let $\ell(w; \xi)$ denote the loss of the model w on datum ξ, and $\mathcal{D}_{\text{val}}$ and $\mathcal{D}_{\text{tra}}$ denote, respectively, the training and validation datasets. Specifically, considering the sought hyper-parameter as the regularization coefficient [28], we aim to solve

$$\begin{aligned} &\min_{\theta \in \mathbb{R}^d} \quad F(\theta) := \mathbb{E}_{\xi \sim \mathcal{D}_{\text{val}}}[\ell(w^*(\theta); \xi)] \\ &\text{s.t.} \quad w^*(\theta) \in \arg\min_{w \in \mathbb{R}^d} \quad \mathbb{E}_{\phi \sim \mathcal{D}_{\text{tra}}}[\ell(w; \phi)] + \sum_{i=1}^{d} \theta_i w_i^2. \end{aligned} \tag{3.3}$$

Model-agnostic meta-learning. The goal of model-agnostic meta-learning (MAML) is to find a common initialization that can adapt to a desired model for new tasks, which inherently consists of two steps: i) training a model over a variety of learning tasks; ii) refining the model for each task. Consider a set of empirically observed tasks collected in $\mathcal{M} := \{1, \ldots, M\}$ drawn from a certain task distribution. Each task m has its local data ξ_m from a certain distribution, which defines its loss function as $F_m(\theta) := \mathbb{E}_{\xi_m}[\ell(\theta; \xi_m)]$, $m \in \mathcal{M}$, where $\theta \in \mathbb{R}^d$ is the parameter of a prediction model (e.g., weights of a neural network), and $\ell(\theta; \xi_m)$ is again the loss on datum ξ_m. As an example, the MAML problem can be formulated as the bilevel problem (3.1), that is [88]

$$\begin{aligned} &\min_{\theta \in \mathbb{R}^d} \; F(\theta) := \frac{1}{M} \sum_{m=1}^{M} F_m\left(y_m^*(\theta)\right) \\ &\text{s. t.} \quad y_m^*(\theta) \in \arg\min_{y_m \in \mathbb{R}^d} \; F_m(y_m) + \frac{\lambda}{2}\|y_m - \theta\|^2, \; \forall m \end{aligned} \tag{3.4}$$

where λ is a constant and $y_m^*(\theta)$ is, initialized with θ, obtained after fine tuning on task m.

3.2 A Single-loop Stochastic Method for Bilevel Problems

In this section, we will first provide background of bilevel problems, and then present our stochastic bilevel gradient method, followed by an ODE analysis to highlight the intuition of our design.

3.2.1 Preliminaries

We use $\|\cdot\|$ to denote the ℓ_2 norm for vectors and Frobenius norm for matrices. We use $\mathcal{F}^k$ to denote the collection of random variables, i.e., $\mathcal{F}^k := \left\{\phi^0, \ldots, \phi^{k-1}, \xi^0, \ldots, \xi^{k-1}\right\}$. For convenience, we define the deterministic version of (3.1) as

$$\min_{\theta \in \mathbb{R}^d} \quad F(\theta) := f\left(\theta, y^*(\theta)\right) \qquad \text{s. t.} \quad y^*(\theta) \in \underset{y \in \mathbb{R}^{d'}}{\arg\min}\, g(\theta, y) \tag{3.5}$$

where the functions are defined as $g(\theta, y) := \mathbb{E}_\phi[g(\theta, y; \phi)]$ and $f(\theta, y) := \mathbb{E}_\xi[f(\theta, y; \xi)]$.

We also define $\nabla^2_{yy} g\left(\theta, y\right)$ as the Hessian matrix of g with respect to y and define $\nabla^2_{\theta y} g\left(\theta, y\right)$ as

$$\nabla^2_{\theta y} g\left(\theta, y\right) := \begin{bmatrix} \frac{\partial^2}{\partial \theta_1 \partial y_1} g\left(\theta, y\right) & \cdots & \frac{\partial^2}{\partial \theta_1 \partial y_{d'}} g\left(\theta, y\right) \\ & \cdots & \\ \frac{\partial^2}{\partial \theta_d \partial y_1} g\left(\theta, y\right) & \cdots & \frac{\partial^2}{\partial \theta_d \partial y_{d'}} g\left(\theta, y\right) \end{bmatrix}.$$

We make the following standard assumptions that are commonly used in stochastic bilevel optimization literature [33, 37, 44].

Assumption 1 (Lipschitz continuity). *For any fixed θ, $\nabla_\theta f(\theta, \cdot)$, $\nabla_y f(\theta, \cdot)$, $\nabla_y g(\theta, y)$, $\nabla^2_{\theta y} g(\theta, \cdot; \phi)$, $\nabla^2_{yy} g(\theta, \cdot; \phi)$ are $L_{f_\theta}, L_{f_y}, L_g, L_{g_{\theta y}}, L_{g_{yy}}$-Lipschitz continuous. For any fixed y, $\nabla_\theta f(\cdot, y; \xi)$, $\nabla_y f(\cdot, y; \xi)$, $\nabla^2_{\theta y} g(\cdot, y; \phi)$, $\nabla^2_{yy} g(\cdot, y; \phi)$ are $\bar{L}_{f_\theta}, \bar{L}_{f_y}, \bar{L}_{g_{\theta y}}, \bar{L}_{g_{yy}}$-Lipschitz continuous.*

Assumption 2 (strong convexity of lower-level objective). *For any fixed θ, $g(\theta, y)$ is μ_g-strongly convex in y, that is, $\nabla^2_{yy} g(\theta, y) \succeq \mu_g I$.*

Assumptions 1 and 2 together ensure that the first- and second-order derivations of $f(\theta, y), g(\theta, y)$ as well as the solution mapping $y^*(\theta)$ are well-behaved.

Assumption 3 (stochastic derivatives). *The stochastic derivatives* $\nabla_\theta f(\theta, y; \xi)$, $\nabla_y f(\theta, y; \xi)$, $\nabla_y g(\theta, y; \phi)$, $\nabla^2_{\theta y} g(\theta, y, \phi)$, *and* $\nabla^2_{yy} g(\theta, y, \phi)$ *are unbiased estimators of* $\nabla_\theta f(\theta, y)$, $\nabla_y f(\theta, y)$, $\nabla_y g(\theta, y)$, $\nabla^2_{\theta y} g(\theta, y)$, *and* $\nabla^2_{yy} g(\theta, y)$, *respectively; and their variances are bounded by* $\sigma^2_{f_\theta}, \sigma^2_{f_y}$, $\sigma^2_{g_y}$, $\sigma^2_{g_{\theta y}}, \sigma^2_{g_{yy}}$, *respectively. Moreover, their monuments are bounded by*

$$\mathbb{E}_\xi[\|\nabla_\theta f(\theta, y; \xi)\|^p] \le C^p_{f_\theta}, \qquad \mathbb{E}_\xi[\|\nabla_y f(\theta, y; \xi)\|^p] \le C^p_{f_y}, \; p = 2, 4 \tag{3.6a}$$

$$\mathbb{E}_\phi[\|\nabla^2_{\theta y} g(\theta, y; \phi)\|^2] \le C^2_{g_{\theta y}}, \quad \mathbb{E}_\phi[\|\nabla^2_{yy} g(\theta, y; \phi)\|^2] \le C^2_{g_{yy}}. \tag{3.6b}$$

Assumption 3 is the counterpart of the unbiasedness and bounded variance assumption in the single-level stochastic optimization. In addition, the bounded moments in Assumption 3 ensure the Lipschitz continuity of the upper-level gradient $\nabla F(\theta)$.

We first highlight the inherent challenge of directly applying the single-level SGD method [93] to the bilevel problem (3.1). To illustrate this point, we derive the gradient of the upper-level function $F(\theta)$ in the next proposition; see the proof in Appendix.

Proposition 1 *Under Assumption 2, we have the gradients*

$$\nabla_\theta y^*(\theta)^\top := -\nabla^2_{\theta y} g(\theta, y^*(\theta)) \left[\nabla^2_{yy} g(\theta, y^*(\theta))\right]^{-1} \tag{3.7a}$$

$$\nabla F(\theta) = \nabla_\theta f(\theta, y^*(\theta)) + \nabla_\theta y^*(\theta)^\top \nabla_y f(\theta, y^*(\theta)). \tag{3.7b}$$

Notice that obtaining an unbiased stochastic estimate of $\nabla F(\theta)$ and applying SGD on θ face two main difficulties: **(D1)** the gradient $\nabla F(\theta)$ at θ depends on the minimizer of the lower-level problem $y^*(\theta)$; **(D2)** even if $y^*(\theta)$ is known, it is hard to apply the stochastic approximation to obtain an unbiased estimate of $\nabla F(\theta)$ since $\nabla F(\theta)$ is nonlinear in $\nabla^2_{yy} g(\theta, y^*(\theta))$; see the discussion of (D2) in stochastic compositional optimization literature, e.g., [117, 11].

Similar to some existing algorithms for bilevel problems, our method addresses (D1) by evaluating $\nabla F(\theta)$ on a certain vector y in place of $y^*(\theta)$, but it differs in how to recursively update y and how to address (D2). Resembling the definition (3.7) with $y^*(\theta)$ replaced by y, we introduce the notation

$$\overline{\nabla}_\theta f\left(\theta, y\right) := \nabla_\theta f\left(\theta, y\right) - \nabla^2_{\theta y} g\left(\theta, y\right)\left[\nabla^2_{yy} g\left(\theta, y\right)\right]^{-1} \nabla_y f\left(\theta, y\right). \tag{3.8}$$

As we will show in Lemma 9 of Appendix, Assumptions 1-3 ensure that $\nabla F(\cdot)$, $\overline{\nabla}_\theta f(\theta, \cdot)$, and $y^*(\cdot)$ are all Lipschitz continuous with constants L_F, L_f, L_y, respectively.

3.2.2 A single-timescale bilevel optimization method

Before we present our method, we first review a successful recent effort. To overcome the difficulty of applying plain-vanilla SGD, a *two-timescale* stochastic approximation (TTSA) algorithm has been recently developed in [37]. TTSA is a single-loop algorithm and amenable to efficient implementation. It consists of two sequences $\{\theta^k\}$ and $\{y^k\}$: for a given θ^k, y^k estimates the minimizer $y^*(\theta^k)$; and, θ^k estimates the minimizer θ^*. For notational brevity, we define

$$h_g^k := \nabla_y g(\theta^k, y^k; \phi^k), \qquad h_{yy}^k(\phi) := \nabla^2_{yy} g(\theta^k, y^k; \phi), \qquad h_{\theta y}^k(\phi) := \nabla^2_{\theta y} g(\theta^k, y^k; \phi). \tag{3.9}$$

With α_k and β_k denoting two sequences of stepsizes, the TTSA recursion is given by

$$y^{k+1} = y^k - \beta_k h_g^k \tag{3.10a}$$

$$\theta^{k+1} = \theta^k - \alpha_k \left(\nabla_\theta f(\theta^k, y^k; \xi^k) - h_{\theta y}^k(\phi^k) \nabla_{yy}^{-1} \nabla_y f(\theta^k, y^k; \xi^k)\right) \tag{3.10b}$$

where ∇_{yy}^{-1} is a mini-batch approximation of $\left[\nabla^2_{yy} g(\theta^k, y^k)\right]^{-1}$. To ensure convergence, TTSA requires y^k to be updated in a timescale faster than that of θ^k so that θ^k is relatively static with respect to y^k; i.e., $\lim_{k\to\infty} \alpha_k/\beta_k = 0$ [37]. However, this prevents TTSA from choosing the stepsize $\mathcal{O}(1/\sqrt{k})$ as SGD, and also results in its *suboptimal complexity.*

We find that the key reason preventing TTSA from using a single-timescale update is its undesired stochastic upper-level gradient estimator (3.10b) that uses an inaccurate lower-level variable y^k to approximate $y^*(\theta^k)$. With more insights given in Section 3.2.3, we propose

Algorithm 3 STABLE for stochastic bilevel problems

```
initialize: $\theta^0, y^0, H^0_{\theta y}, H^0_{yy}$, stepsizes $\{\alpha_k, \beta_k\}$.
for $k = 0, 1, \ldots, K-1$ do
    compute $h^{k-1}_{\theta y}(\phi^k)$ and $h^k_{\theta y}(\phi^k)$    ▷ randomly select datum $\phi^k$
    update $H^k_{\theta y}$ via (3.12a)
    compute $h^{k-1}_{yy}(\phi^k)$ and $h^k_{yy}(\phi^k)$
    update $H^k_{yy}$ via (3.12b)
    compute $\nabla_\theta f\left(\theta^k, y^k; \xi^k\right)$, $\nabla_y f\left(\theta^k, y^k; \xi^k\right)$    ▷ randomly select datum $\xi^k$
    update $\theta^k$ and $y^k$ via (3.11)
end for
```

a new stochastic bilevel optimization method based on a new stochastic bilevel gradient estimator, which we term Single-Timescale stochAstic BiLevEl optimization (**STABLE**) method. Its recursion is given by

$$\theta^{k+1} = \theta^k - \alpha_k \left(\nabla_\theta f(\theta^k, y^k; \xi^k) - H^k_{\theta y}(H^k_{yy})^{-1}\nabla_y f(\theta^k, y^k; \xi^k)\right) \tag{3.11a}$$

$$y^{k+1} = y^k - \beta_k h^k_g - (H^k_{yy})^{-1}(H^k_{\theta y})^\top(\theta^{k+1} - \theta^k). \tag{3.11b}$$

In (3.11), the estimates of second-order derivatives are updated as (with stepsize $\tau_k > 0$)

$$H^k_{\theta y} = \overline{\mathcal{P}}\left((1-\tau_k)\Big(H^{k-1}_{\theta y} - h^{k-1}_{\theta y}(\phi^k)\Big) + h^k_{\theta y}(\phi^k)\right) \tag{3.12a}$$

$$H^k_{yy} = \underline{\mathcal{P}}\left((1-\tau_k)\Big(H^{k-1}_{yy} - h^{k-1}_{yy}(\phi^k)\Big) + h^k_{yy}(\phi^k)\right) \tag{3.12b}$$

where $\overline{\mathcal{P}}$ is the projection to set $\{X : \|X\| \leq C_{g_{\theta y}}\}$ and $\underline{\mathcal{P}}$ is the projection to $\{X : X \succeq \mu_g I\}$.

Compared with (3.10) and other existing algorithms, the unique features of STABLE lie in: **(F1)** its y^k-update that will be shown to better "predict" the next $y^*(\theta^{k+1})$; and, **(F2)** a recursive update of $H^k_{\theta y}, H^k_{yy}$ that is motivated by the advanced variance reduction techniques for single-level problems [81, 15] and the recent stochastic compositional optimization method [11]. The marriage of (F1)-(F2) enables STABLE to have a better estimate of $\nabla f(\theta^k)$, which is responsible for its improved convergence. Note that we use three stepsizes α_k, β_k and τ_k in

(3.11), but all of them decrease at the same rate as SGD. As we will show later, for a class of bilevel problems, the single-timescale recursion (3.11) achieves the same convergence rate as SGD for single-level problems. See a summary of STABLE in Algorithm 3.

Remark 1 *Note that the projection in* (3.12a) *is not uncommon in stochastic algorithms to ensure stability, and the eigenvalue truncation in* (3.12b) *is a usual subroutine in Newton-based methods, which is also referred to the positive definite truncation [83].*

3.2.3 Continuous-time ODE analysis

Similar to the stochastic compositional optimization [11], we provide some intuition of our algorithm design via an ODE for the deterministic problem (3.5). To minimize $F(\theta)$, we use an ODE analysis to design a continuous dynamic

$$\dot{\theta}(t) = -\alpha\mathcal{T}(\theta(t), y(\theta(t))) \tag{3.13}$$

by choosing an operator $\mathcal{T}$. For single-level minimization of a smooth function $h(\theta(t))$, one can use the gradient flow $\dot{\theta}(t) = -\alpha\nabla h(\theta(t))$. For bilevel minimization (3.5), however, we shall avoid $\mathcal{T}(\theta, y) = \nabla_\theta\left(f(\theta, y^*(\theta)\right)$ and instead use y to approximate $y^*(\theta)$. Here note that we have dropped (t) for conciseness. Hence, define the operator as

$$\mathcal{T}(\theta, y) := \nabla_\theta f(\theta, y) - \nabla^2_{\theta y} g(\theta, y)[\nabla^2_{yy} g(\theta, y)]^{-1}\nabla_y f(\theta, y) \overset{(3.8)}{=} \overline{\nabla}_\theta f(\theta, y). \tag{3.14}$$

Here, the variable y follows another dynamic that we specify below, which accompanies the θ-dynamic (3.13). We will also find a corresponding Lyapunov function V such that

$$\text{(C1) } \dot{V} < 0; \quad \text{and,} \quad \text{(C2) } \dot{V} = 0 \text{ if and only if } \nabla F(\theta) = 0 \text{ and } y = y^*(\theta).$$

If the $\dot{\theta}$ and $\dot{y}$ dynamics drive an appropriate Lyapunov function V satisfying (C1) and (C2), then θ converges to a stationary point of the upper-level problem $F(\theta)$ and y converges to the solution of the lower-level problem.

We first state the results for the continuous-time dynamics below.

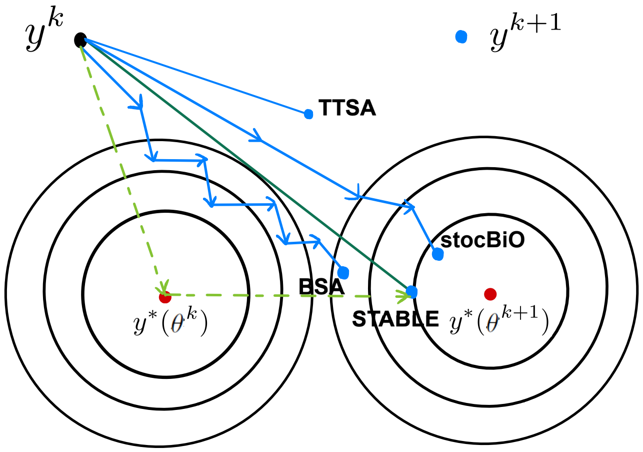

Figure 3.1: A geometric illustration of the y^k update under different algorithms; black dot represents y^k, red dots represent the lower-level solution $y^*(\theta^k)$ and $y^*(\theta^{k+1})$, blue dots represent y^{k+1} under different algorithms, and blue arrow denotes the inner loop updates. **STABLE** updates y^k by linearly combining the stochastic gradient direction towards $y^*(\theta^k)$ and the moving direction from $y^*(\theta^k)$ to $y^*(\theta^{k+1})$. In contrast, **BSA** [33] runs multiple stochastic gradient steps; **TTSA** [37] runs one stochastic gradient step with a smaller stepsize; **stocBiO** [44] runs multiple stochastic gradient steps with an increasing batch size.

Theorem 1 (Continuous-time dynamics) *If we define the θ- and y-dynamics as*

$$\dot{\theta} = -\alpha \nabla_\theta f(\theta, y) - \alpha \nabla^2_{\theta y} g(\theta, y)[\nabla^2_{yy} g(\theta, y)]^{-1} \nabla_y f(\theta, y) \tag{3.15a}$$

$$\dot{y} = -\beta \nabla_y g(\theta, y) - \left[\nabla^2_{yy} g(\theta, y)\right]^{-1} \nabla^2_{\theta y} g(\theta, y)\, \dot{\theta} \tag{3.15b}$$

and choose the constants α and β appropriately, then there exists a Lyapunov function V of the θ- and y-dynamics that satisfies (C1) and (C2).

Proof: To highlight the intuition, we provide a constructive proof of this theorem. We first try $V_0 := f(\theta, y^*(\theta))$. To clarify, we can use $y^*(\theta)$ in a Lyapunov function but not in a dynamic to evolve a quantity. In this case, we have

$$\begin{aligned}\dot{V}_0 &= \langle \nabla_\theta f(\theta, y^*(\theta)), \dot{\theta}\rangle + \langle \nabla_y f(\theta, y^*(\theta)), \nabla_\theta y^*(\theta)\dot{\theta}\rangle \\ &= \langle \nabla_\theta f(\theta, y^*(\theta)) + \nabla_\theta y^*(\theta)^\top \nabla_y f(\theta, y^*(\theta)), \dot{\theta}\rangle.\end{aligned}$$

Recall the definition in (3.7). Then we have

$$\begin{aligned}\dot{V}_0 &= -\alpha\langle \mathcal{T}(\theta, y^*(\theta)), \mathcal{T}(\theta, y)\rangle \\ &\overset{(a)}{\leq} -\alpha\|\mathcal{T}(\theta, y^*(\theta))\|^2 + \alpha\|\overline{\nabla}_\theta f(\theta, y) - \overline{\nabla}_\theta f(\theta, y^*(\theta))\|\|\mathcal{T}(\theta, y^*(\theta))\| \\ &\overset{(b)}{\leq} -\alpha\|\mathcal{T}(\theta, y^*(\theta))\|^2 + \alpha L_f\|y - y^*(\theta)\|\|\mathcal{T}(\theta, y^*(\theta))\| \\ &\overset{(c)}{\leq} -\frac{\alpha}{2}\|\mathcal{T}(\theta, y^*(\theta))\|^2 + \frac{\alpha L_f^2}{2}\|y - y^*(\theta)\|^2 \qquad (3.16)\end{aligned}$$

where (a) uses the Cauchy-Schwarz inequality, (b) follows from the L_f-Lipschitz continuity of $\overline{\nabla}_\theta f(\theta, \cdot)$ established in Lemma 9, and (c) is due to the Young's inequality.

To satisfy (C1), we have $\dot{V}_0 \leq 0$ only if $L_f\|y - y^*(\theta)\| \leq \|\mathcal{T}(\theta, y^*(\theta))\|$, thus, requiring the information of $\|y - y^*(\theta)\|$ — not doable without knowing $y^*(\theta)$.

Let us try to mitigate the term $\|y(\theta) - y^*(\theta)\|^2$ by defining the following new Lyapunov function:

$$V := V_0 + \frac{1}{2}\|y - y^*(\theta)\|^2 = f(\theta, y^*(\theta)) + \frac{1}{2}\|y - y^*(\theta)\|^2 \qquad (3.17)$$

which implies that

$$\begin{aligned}\dot{V} &= -\alpha\langle \mathcal{T}(\theta, y^*(\theta)), \mathcal{T}(\theta, y)\rangle + \langle y - y^*(\theta), \dot{y} - \nabla_\theta y^*(\theta)\dot{\theta}\rangle \\ &\overset{(3.16)}{\leq} -\frac{\alpha}{2}\|\mathcal{T}(\theta, y^*(\theta))\|^2 + \frac{\alpha L_f^2}{2}\|y - y^*(\theta)\|^2 + \langle y - y^*(\theta), \dot{y} - \nabla_\theta y^*(\theta)\dot{\theta}\rangle \qquad (3.18) \\ &\leq -\frac{\alpha}{2}\|\mathcal{T}(\theta, y^*(\theta))\|^2 - \left(\beta - \frac{\alpha L_f^2}{2}\right)\|y - y^*(\theta)\|^2 \\ &\quad + \langle y - y^*(\theta), \dot{y} + \beta(y - y^*(\theta)) - \nabla_\theta y^*(\theta)\dot{\theta}\rangle \qquad (3.19)\end{aligned}$$

where $\beta > 0$ is a fixed constant. The first two terms in the RHS of (3.19) are non-positive given that $\alpha \geq 0$ and $\beta \geq \alpha L_f^2/2$, but the last term can be either positive or negative. To control the last term and thus ensure the descent of $V(t)$, we are motivated to use a y-dynamic like

$$\dot{y} \approx -\beta(y - y^*(\theta)) + \nabla_\theta y^*(\theta)\dot{\theta}. \tag{3.20}$$

To avoid using y^* in a dynamic, we approximate $y - y^*(\theta)$ by $\nabla_y g(\theta, y)$ and $\nabla_\theta y^*(\theta)$ by (cf. (3.7a))

$$\nabla_\theta y(\theta) := -\left[\nabla^2_{yy} g\left(\theta, y\right)\right]^{-1} \nabla^2_{\theta y} g\left(\theta, y\right). \tag{3.21}$$

These choices lead to the y-dynamics:

$$\dot{y} = -\beta \nabla_y g(\theta, y) + \nabla_\theta y(\theta)\dot{\theta}. \tag{3.22}$$

Although we approximate (3.20) by (3.22), next we will plug y-dynamics (3.22) into (3.19) and show that V satisfies (C1). Specifically, plugging (3.22) into (3.18) leads to

$$\langle y - y^*(\theta), \dot{y} - \nabla_\theta y^*(\theta)\dot{\theta}\rangle = -\langle y - y^*(\theta), \beta\nabla_y g(\theta, y) - \nabla_\theta y(\theta)\dot{\theta} + \nabla_\theta y^*(\theta)\dot{\theta}\rangle. \tag{3.23}$$

As $g(\theta, \cdot)$ is μ_g-strongly convex by Assumption 2, we have

$$\left\langle y - y^*(\theta), \nabla_y g(\theta, y) - \nabla_y g(\theta, y^*(\theta))\right\rangle \geq \mu_g \|y - y^*(\theta)\|^2 \tag{3.24}$$

where $\nabla_y g(\theta, y^*(\theta)) = 0$ as $y^*(\theta)$ minimizes $g(\theta, y)$.

Therefore, plugging (3.24) into (3.23), we have

$$\begin{aligned} \langle y - y^*(\theta), \dot{y} - \nabla_\theta y^*(\theta)\dot{\theta}\rangle &\leq -\langle y - y^*(\theta), (\nabla_\theta y^*(\theta) - \nabla_\theta y(\theta))\dot{\theta}\rangle - \beta\mu_g\|y - y^*(\theta)\|^2 \\ &\leq \|y - y^*(\theta)\|\|\nabla_x y^*(\theta) - \nabla_\theta y(\theta)\|\|\dot{\theta}\| - \beta\mu_g\|y - y^*(\theta)\|^2 \\ &\leq \alpha B_\theta L_y\|y - y^*(\theta)\|^2 - \beta\mu_g\|y - y^*(\theta)\|^2 \end{aligned} \tag{3.25}$$

where the second inequality uses the Cauchy-Schwarz inequality, and the last inequality follows the bound B_θ of $\|\dot{\theta}\|$ and the Lipschitz constant L_y of $\nabla_\theta y(\theta)$, both of which can be derived from Assumptions 1–3.

Now plugging (3.25) into (3.18), we have

$$\dot{V} \leq -\frac{\alpha}{2}\|\mathcal{T}(\theta, y^*(\theta))\|^2 - \left(\beta\mu_g - \frac{\alpha L_f^2}{2} - \alpha B_\theta L_y\right)\|y - y^*(\theta)\|^2. \tag{3.26}$$

Now let us check (C1) and (C2). To ensure $\dot{V} \leq 0$ in (C1), we can set $\alpha \leq \frac{2\mu_g\beta}{L_f^2 + 2B_\theta L_y}$. For (C2), we have $\dot{V} = 0$ if and only if $y = y^*(\theta)$ and $\mathcal{T}(\theta, y^*(\theta)) = \nabla F(\theta) = 0$.

With the insights gained from the continuous-time update (3.15), our stochastic update (3.11) essentially discretizes time t into iteration k, and replaces the first- and second-order derivatives in $\dot{\theta}$ and $\dot{y}$ by their recursive (variance-reduced) stochastic values in (3.12).

Remark 2 *The key ingredient of our STABLE method is the design of the lower-level update on y^k, which leads to a more accurate stochastic estimate of $\nabla f(\theta^k)$. See a comparison of the y-update with other algorithms in Figure 3.1. In the update* (3.11)*, we implement the SGD-like update for the upper-level variable θ^k. With the lower-level y^k update unchanged, it is easy to apply SGD-improvement techniques such as momentum and variance reduction, to accelerate the convergence of STABLE. This will help STABLE achieve state-of-the-art performance for stochastic bilevel optimization.*

3.3 Convergence Analysis

In this section, we establish the convergence rate of our single-timescale STABLE algorithm. We will highlight the key steps of the proof and leave the detailed analysis in Appendix.

3.3.1 Main results

We first present the result of our algorithm when the upper-level function $F(\theta)$ is nonconvex in θ.

Theorem 2 (Nonconvex) *Under Assumptions 1–3, if we choose the stepsizes as*

$$\beta_k \leq \min\left\{\frac{1}{\sqrt{K}}, \frac{\mu_g/L_g}{32(\mu_g+L_g)}\right\} \tag{3.27a}$$

$$\alpha_k \leq \min\left\{\beta_k, \frac{(c+2C_{g\theta y}^2 C_{f_y}^2/\mu_g^4)^{-1}}{\sqrt{K}}, \frac{(c+2C_{f_y}^2/\mu_g^2)^{-1}}{\sqrt{K}}, \frac{\mu_g L_g \beta_k/(\mu_g+L_g)}{2(c+L_f^2)}\right\} \tag{3.27b}$$

and $\tau_k = \frac{1}{\sqrt{K}}$, then the iterates $\{\theta^k\}$ and $\{y^k\}$ satisfy

$$\mathbb{E}\left[\left\|\nabla f(\theta^K)\right\|^2\right] = \mathcal{O}\left(\frac{1}{\sqrt{K}}\right) \quad \textit{and} \quad \mathbb{E}\left[\left\|y^K - y^*(\theta^k)\right\|^2\right] = \mathcal{O}\left(\frac{1}{\sqrt{K}}\right) \tag{3.28}$$

where $y^*(\theta^k)$ is the minimizer of the lower-level problem in (3.1b), *and $c > 0$ is an absolute constant that is independent of the stepsizes $\alpha_k, \beta_k, \tau_k$ and the number of iterations K.*

Theorem 2 implies that the convergence rate of STABLE to the stationary point of (3.1) is $\mathcal{O}(K^{-\frac{1}{2}})$. Since each iteration of STABLE only uses two samples (see Algorithm 3), the sample complexity to achieve an ϵ-stationary point of (3.1) is $\mathcal{O}(\epsilon^{-2})$, which is on the same order of SGD's sample complexity for the single-level nonconvex problems [29], and significantly improves the state-of-the-art single-loop TTSA's convergence rate $\mathcal{O}(\epsilon^{-2.5})$ [37]. In addition, this convergence rate is not directly comparable to other recently developed bilevel optimization methods, e.g., [33, 44] since STABLE does not need the increasing batchsize nor double-loop. Regarding the sample complexity, however, STABLE improves over [33, 44] by at least the order of $\mathcal{O}(\log(\epsilon^{-1}))$.

We next present the result in the strongly convex case. For the strong-convex case, we slightly modify the update of θ^k in (3.11a) to

$$\theta^{k+1} = \mathcal{P}_\Theta\left(\theta^k - \alpha_k\left(\nabla_\theta f(\theta^k, y^k; \xi^k) - H_{\theta y}^k (H_{yy}^k)^{-1} \nabla_y f(\theta^k, y^k; \xi^k)\right)\right) \tag{3.29}$$

where $\mathcal{P}_\Theta$ denotes the projection on set Θ.

We need the following additional assumption.

Assumption 4 (strong convexity). *Function $F(\theta)$ is μ-strongly convex in θ, that is, $\nabla_{xx}^2 F(\theta) \succeq \mu I$.*

Theorem 3 (Strongly convex) *Under Assumptions 1–4, if we choose the stepsizes as*

$$\beta_k = \tau_k \leq \min\left\{\frac{\mu_g/L_g}{32(\mu_g + L_g)}, \frac{1}{K_0 + k}\right\} \tag{3.30a}$$

$$\alpha_k \leq \min\left\{\sqrt{\frac{\mu_g L_g}{4c(\mu_g + L_g)}}, \frac{\mu\mu_g L_g}{2L_f^2(\mu_g + L_g)}, \frac{1}{\sqrt{4c}}, \frac{\mu\mu_g^4}{8C_{g_{\theta y}}^2 C_{f_y}^2}, \frac{\mu\mu_g^2}{8C_{f_y}^2}\right\}\beta_k \tag{3.30b}$$

where $K_0 > 0$ is a sufficiently large constant and $c > 0$ is an absolute constant that is independent of the stepsizes $\alpha_k, \beta_k, \tau_k$, then the iterates $\{\theta^k\}$ and $\{y^k\}$ satisfy

$$\mathbb{E}\left[\left\|\theta^k - \theta^*\right\|^2\right] = \mathcal{O}\left(\frac{1}{k}\right) \quad \textit{and} \quad \mathbb{E}\left[\left\|y^k - y^*(\theta^k)\right\|^2\right] = \mathcal{O}\left(\frac{1}{k}\right) \tag{3.31}$$

where the solution θ^* is defined as $\theta^* = \arg\min_{\theta\in\times} F(\theta)$ and $y^*(\theta^k)$ is the minimizer of the lower-level problem in (3.1b)*.*

Theorem 3 implies that to achieve an ϵ-optimal solution for both the lower-level and upper-level problems, the sample complexity of STABLE is $\mathcal{O}(\epsilon^{-1})$. This complexity is on the same order of SGD's complexity for the single-level strongly convex problems [29], and improves the state-of-the-art single-loop TTSA's sample complexity $\mathcal{O}(\epsilon^{-2})$ for an ϵ-optimal upper-level solution and $\mathcal{O}(\epsilon^{-1.5})$ for an ϵ-optimal lower-level solution [37]. Compared with double-loop bilevel algorithms in this strong-convex case, STABLE also improves over the BSA's query complexity $\mathcal{O}(\epsilon^{-1})$ in terms of the stochastic upper-level function and $\mathcal{O}(\epsilon^{-2})$ in terms of the stochastic lower-level function [33].

3.3.2 Proof sketch

Next we highlight the key steps of the proof towards Theorem 2. The proof for the strongly convex case in Theorem 3 will follow similar steps.

For simplicity of the convergence analysis, we define the following Lyapunov function

$$\mathbb{V}^k := f(\theta^k) + \|y^k - y^*(\theta^k)\|^2 + \|H_{yy}^k - \nabla_{yy}^2 g(\theta^k, y^k)\|^2 + \|H_{\theta y}^k - \nabla_{\theta y}^2 g(\theta^k, y^k)\|^2 \tag{3.32}$$

which mimics the continuous-time Lyapunov function (3.17) for the deterministic problem.

Similar to the ODE analysis, we first quantify the difference between two Lyapunov functions as

$$\begin{aligned}\mathbb{V}^{k+1} - \mathbb{V}^{k} =& \underbrace{f(\theta^{k+1}) - f(\theta^{k})}_{\text{Lemma 6}} + \underbrace{\|y^{k+1} - y^{*}(\theta^{k+1})\|^{2} - \|y^{k} - y^{*}(\theta^{k})\|^{2}}_{\text{Lemma 7}} \\ &+ \underbrace{\|H_{yy}^{k+1} - \nabla_{yy}^{2} g(\theta^{k+1}, y^{k+1})\|^{2} - \|H_{yy}^{k} - \nabla_{yy}^{2} g(\theta^{k}, y^{k})\|^{2}}_{\text{Lemma 8}} \\ &+ \underbrace{\|H_{\theta y}^{k+1} - \nabla_{\theta y}^{2} g(\theta^{k+1}, y^{k+1})\|^{2} - \|H_{\theta y}^{k} - \nabla_{\theta y}^{2} g(\theta^{k}, y^{k})\|^{2}}_{\text{Lemma 8}}. \end{aligned} \tag{3.33}$$

The difference in (3.33) consists of four difference terms: the first term quantifies the descent of the upper-level objective functions; the second term characterizes the descent of the lower-level optimization errors; and, the third and fourth terms measure the estimation error of the second-order quantities. We will bound them, respectively, in the ensuing lemmas.

We will first analyze the descent of the upper-level objective in the next lemma.

Lemma 6 (Descent of upper level) *Suppose Assumptions 1–3 hold. The sequence of* θ^k *satisfies*

$$\begin{aligned}\mathbb{E}[f(\theta^{k+1})] - \mathbb{E}[f(\theta^{k})] \leq& -\frac{\alpha_k}{2}\mathbb{E}[\|\nabla f(\theta^{k})\|^{2}] + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^{k}\|^{2}] + \alpha_k L_f^2 \mathbb{E}[\|y^{k} - y^{*}(\theta^{k})\|^{2}] \\ &+ \frac{2C_{g_{\theta y}}^{2} C_{f_y}^{2} \alpha_k}{\mu_g^4}\mathbb{E}[\|H_{yy}^{k} - \nabla_{yy}^{2} g(\theta^{k}, y^{k})\|^{2}] \\ &+ \frac{2C_{f_y}^{2}\alpha_k}{\mu_g^2}\mathbb{E}[\|H_{\theta y}^{k} - \nabla_{\theta y}^{2} g(\theta^{k}, y^{k})\|^{2}] \end{aligned} \tag{3.34}$$

where L_f, L_F *are defined in Lemma 9 of Appendix, and* $C_{g_{\theta y}}$ *is the projection radius in* (3.12a)*.*

Lemma 6 implies that the descent of the upper-level objective functions depends on the error of the lower-level variable y^k, and the estimation errors of H_{yy}^k and $H_{\theta y}^k$. We will next analyze the error of the lower-level variable, which is the key step to improving the existing results.

Lemma 7 (Error of lower level) *Suppose that Assumptions 1–3 hold, and y^{k+1} is generated by running iteration* (3.11) *given θ^k. If we choose $\beta_k \leq \frac{2}{\mu_g+L_g}$, then y^{k+1} satisfies*

$$\mathbb{E}\left[\|y^*(\theta^{k+1}) - y^{k+1}\|^2|\mathcal{F}^k\right] \leq \left(1 - \frac{\mu_g L_g \beta^k}{\mu_g + L_g} + \frac{c\alpha_k^2}{\beta_k}\right)\|y^k - y^*(\theta^k)\|^2 + \left(1 + \frac{\mu_g L_g \beta^k}{\mu_g + L_g}\right)\beta_k^2 \sigma_{g_y}^2$$
$$+ \frac{c\alpha_k^4}{\beta_k} + \mathbb{E}\left[\|H_{yy}^k - \nabla_{yy}^2 g(\theta^k, y^k)\|^2|\mathcal{F}^k\right]\frac{c\alpha_k^2}{\beta_k}$$
$$+ \mathbb{E}\left[\|H_{\theta y}^k - \nabla_{\theta y}^2 g(\theta^k, y^k)\|^2|\mathcal{F}^k\right]\frac{c\alpha_k^2}{\beta_k}. \tag{3.35}$$

Roughly speaking, Lemma 7 implies that if the stepsizes α_k^2 and β_k^2 and the estimation errors of H_{yy}^k and $H_{\theta y}^k$ are decreasing fast enough, the error of y^{k+1} will also decrease.

Since the RHS of both Lemmas 6 and 7 critically depend on the quality of H_{yy}^k and $H_{\theta y}^k$, we will next build upon the results in [11, Lemma 2] to analyze the estimation errors.

Lemma 8 (Estimation errors of $H_{\theta y}^k$ and H_{yy}^k) *Suppose Assumptions 1–3 hold, and $H_{\theta y}^k$ and H_{yy}^k are generated by running* (3.12). *The mean square error of $H_{\theta y}^k$ satisfies*

$$\mathbb{E}\Big[\|H_{\theta y}^k - \nabla_{\theta y}^2 g(\theta^k, y^k)\|^2 \mid \mathcal{F}^k\Big] \leq (1-\tau_k)^2\|H_{\theta y}^{k-1} - \nabla_{\theta y}^2 g(\theta^{k-1}, y^{k-1})\|^2 + 2\tau_k^2\sigma_{g_{\theta y}}^2$$
$$+ 2(1-\tau_k)^2(\bar{L}_{g_{\theta y}}^2 + L_{g_{\theta y}}^2)\|\theta^k - \theta^{k-1}\|^2 + 2(1-\tau_k)^2(\bar{L}_{g_{\theta y}}^2 + L_{g_{\theta y}}^2)\|y^k - y^{k-1}\|^2 \tag{3.36}$$

where the constants $L_{g_{\theta y}}, L_{g_{yy}}, \bar{L}_{g_{\theta y}}, \bar{L}_{g_{yy}}, \sigma_{g_{\theta y}}, \sigma_{g_{yy}}$ are defined in Assumptions 1 and 3. And likewise, the mean square error of H_{yy}^k satisfies

$$\mathbb{E}\Big[\|H_{yy}^k - \nabla_{yy}^2 g(\theta^k, y^k)\|^2 \mid \mathcal{F}^k\Big] \leq (1-\tau_k)^2\|H_{yy}^{k-1} - \nabla_{yy}^2 g(\theta^{k-1}, y^{k-1})\|^2 + 2\tau_k^2\sigma_{g_{yy}}^2$$
$$+ 2(1-\tau_k)^2(\bar{L}_{g_{yy}}^2 + L_{g_{yy}}^2)\|\theta^k - \theta^{k-1}\|^2 + 2(1-\tau_k)^2(\bar{L}_{g_{yy}}^2 + L_{g_{yy}}^2)\|y^k - y^{k-1}\|^2. \tag{3.37}$$

Intuitively, the update of θ^k is bounded and so is the update of y^k, and thus $\|\theta^k - \theta^{k-1}\|^2 = \mathcal{O}(\alpha_{k-1}^2)$ and $\|y^k - y^{k-1}\|^2 = \mathcal{O}(\beta_{k-1}^2)$. Plugging them into the RHS of Lemma 8, it suggests that if the stepsizes $\alpha_k^2, \beta_k^2, \tau_k^2$ are decreasing, then the estimation errors of $H_{\theta y}^k$ and H_{yy}^k also decrease.

Applying Lemmas 6–8 to (3.33) and rearranging terms, we will be able to get

$$\mathbb{E}[\mathbb{V}^{k+1}] - \mathbb{E}[\mathbb{V}^k] \leq -c_1\mathbb{E}[\|y^k - y^*(\theta^k)\|^2] - c_2\mathbb{E}[\|\nabla f(\theta^k)\|^2] + c_3 \tag{3.38}$$

where the constants are $c_1 = \mathcal{O}(\beta_k)$, $c_2 = \mathcal{O}(\alpha_k)$ and $c_3 = \mathcal{O}(\alpha_k^2 + \beta_k^2 + \tau_k^2)$. By choosing stepsizes $\alpha_k, \beta_k, \tau_k$ as (3.27) and telescoping both sides of (3.38), we obtain the main results in Theorem 2.

3.4 Appendix

3.4.1 Auxiliary Lemmas

In this appendix, we first present some auxiliary lemmas that will be used frequently in the proof.

Lemma 9 ([33, Lemma 2.2]) *Under Assumptions 1 and 2, we have*

$$\|\overline{\nabla}_\theta f(\theta, y^*(\theta)) - \overline{\nabla}_\theta f(\theta, y)\| \leq L_f\|y^*(\theta) - y\| \tag{3.39a}$$

$$\|\nabla f(\theta_1) - \nabla f(\theta_2)\| \leq L_F\|\theta_1 - \theta_2\| \tag{3.39b}$$

$$\|y^*(\theta_1) - y^*(\theta_2)\| \leq L_y\|\theta_1 - \theta_2\| \tag{3.39c}$$

and the constants L_f, L_y, L_F are defined as

$$L_f := L_{f_\theta} + \frac{C_{g_{\theta y}} L_{f_y}}{\mu_g} + \frac{C_{f_y}}{\mu_g}\left(L_{f\theta y} + \frac{C_{g_{\theta y}} L_{g_{yy}}}{\mu_g}\right), \quad L_y := \frac{C_{g_{\theta y}}}{\mu_g}$$
$$L_F := \bar{L}_{f_\theta} + \frac{C_{g_{\theta y}}(\bar{L}_{f_y} + L_f)}{\mu_g} + \frac{C_{f_y}}{\mu_g}\left(\bar{L}_{f\theta y} + \frac{C_{g_{\theta y}} \bar{L}_{g_{yy}}}{\mu_g}\right)$$

where the constants are defined in Assumptions 1–3.

3.4.2 Proof of Proposition 1

Proof: Define the Jacobian matrix

$$\nabla_\theta y(\theta) = \begin{bmatrix} \frac{\partial}{\partial \theta_1} y_1(\theta) & \cdots & \frac{\partial}{\partial \theta_d} y_1(\theta) \\ & \cdots & \\ \frac{\partial}{\partial \theta_1} y_{d'}(\theta) & \cdots & \frac{\partial}{\partial \theta_d} y_{d'}(\theta) \end{bmatrix}.$$

By the chain rule, it follows that

$$\nabla F(\theta) := \nabla_\theta f\left(\theta, y^*(\theta)\right) + \nabla_\theta y^*(\theta)^\top \nabla_y f\left(\theta, y^*(\theta)\right). \tag{3.40}$$

The minimizer $y^*(\theta)$ satisfies

$$\nabla_y g(\theta, y^*(\theta)) = 0, \quad \text{thus} \quad \nabla_\theta \left(\nabla_y g(\theta, y^*(\theta))\right) = 0. \tag{3.41}$$

By the chain rule again, it follows that

$$\nabla^2_{\theta y} g\left(\theta, y^*(\theta)\right) + \nabla_\theta y^*(\theta)^\top \nabla^2_{yy} g\left(\theta, y^*(\theta)\right) = 0.$$

By Assumption 2, $\nabla^2_{yy} g\left(\theta, y^*(\theta)\right)$ is invertible, so

$$\nabla_\theta y^*(\theta)^\top := -\nabla^2_{\theta y} g\left(\theta, y^*(\theta)\right) \left[\nabla^2_{yy} g\left(\theta, y^*(\theta)\right)\right]^{-1}. \tag{3.42}$$

By substituting (3.42) into (3.40), we arrive at (3.7).

3.4.3 Proof of Lemma 6

Proof: Now we turn to analyze the update of θ. For convenience, we define the update in (3.11a) as

$$\theta^{k+1} = \theta^k - \alpha_k \bar{h}_f^k \qquad \text{with} \qquad \bar{h}_f^k := \nabla_\theta f\left(\theta^k, y^k; \xi^k\right) - H^k_{\theta y}(H^k_{yy})^{-1} \nabla_y f\left(\theta^k, y^k; \xi^k\right) \tag{3.43}$$

Using the smoothness of $f(\theta^k)$ obtained from Lemma 9, we have

$$\begin{aligned}
&\mathbb{E}[f(\theta^{k+1})|\mathcal{F}^k]\\
&\leq f(\theta^k) + \mathbb{E}[\langle \nabla f(\theta^k), \theta^{k+1} - \theta^k\rangle|\mathcal{F}^k] + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k]\\
&= f(\theta^k) - \alpha_k\langle \nabla f(\theta^k), \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k]\\
&= f(\theta^k) - \alpha_k\|\nabla f(\theta^k)\|^2 + \alpha_k\langle \nabla f(\theta^k), \nabla f(\theta^k) - \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k]\\
&\leq f(\theta^k) - \left(\alpha_k - \frac{\alpha_k^2}{4\gamma_k}\right)\|\nabla f(\theta^k)\|^2 + \gamma_k\|\nabla f(\theta^k) - \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\|^2 + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k]
\end{aligned}\tag{3.44}$$

where the last inequality uses Young's inequality with parameter γ_k. We choose $\gamma_k = \alpha_k/2$.

The approximation error of $\bar{h}_f^k$ can be bounded by

$$\big\|\nabla f(\theta^k) - \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\big\|^2 \tag{3.45}$$

$$\begin{aligned}
&\leq 2\big\|\nabla f(\theta^k) - \overline{\nabla} f(\theta^k, y^k)\big\|^2 + 2\mathbb{E}\big[\|\overline{\nabla} f(\theta^k, y^k) - \mathbb{E}_{\xi^k}[\bar{h}_f^k]\|^2|\mathcal{F}^k\big]\\
&\overset{(a)}{\leq} 2L_f^2\big\|y^k - y^*(\theta^k)\big\|^2 + 2\mathbb{E}\big[\|\overline{\nabla} f(\theta^k, y^k) - \mathbb{E}_{\xi^k}[\bar{h}_f^k]\|^2|\mathcal{F}^k\big]\\
&\overset{(b)}{\leq} 2L_f^2\big\|y^k - y^*(\theta^k)\big\|^2 + 2\big\|(H_{yy}^k)^{-1}H_{\theta y}^k - H_{yy}(\theta^k, y^k)^{-1}H_{\theta y}(\theta^k, y^k)\big\|^2\big\|\nabla_y f(\theta^k, y^k)\big\|^2\\
&\overset{(c)}{\leq} 2L_f^2\|y^k - y^*(\theta^k)\|^2 + \frac{4C_{g_{\theta y}}^2 C_{f_y}^2}{\mu_g^4}\mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2|\mathcal{F}^k] + \frac{4C_{f_y}^2}{\mu_g^2}\mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2|\mathcal{F}^k]
\end{aligned}$$

where (a) follows from Lemma 9, (b) uses the fact that

$$\mathbb{E}_{\xi^k}[\bar{h}_f^k|\mathcal{F}^k] = \nabla_\theta f\left(\theta^k, y^k\right) - (H_{yy}^k)^{-1}H_{\theta y}^k\nabla_y f\left(\theta^k, y^k\right) \tag{3.46}$$

and (c) follows the same steps of (3.56) and Assumption 3. Plugging (3.45) into (3.44) and taking expectation over all the randomness lead to the lemma.

3.4.4 Proof of Lemma 7

Proof: We start by decomposing the error of the lower level variable as

$$\begin{aligned}
&\mathbb{E}\left[\|y^{k+1}-y^*(\theta^{k+1})\|^2|\mathcal{F}^k\right]\\
&=\mathbb{E}\left[\|y^k-\beta_k h_g^k-y^*(\theta^k)+y^*(\theta^k)-y^*(\theta^{k+1})-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top(\theta^{k+1}-\theta^k)\|^2|\mathcal{F}^k\right]\\
&\leq(1+\varepsilon)\underbrace{\mathbb{E}[\|y^k-\beta_k h_g^k-y^*(\theta^k)\|^2|\mathcal{F}^k]}_{I_1}\\
&\quad+(1+\varepsilon^{-1})\underbrace{\mathbb{E}[\|y^*(\theta^k)-y^*(\theta^{k+1})-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top(\theta^{k+1}-\theta^k)\|^2|\mathcal{F}^k]}_{I_2}. \qquad (3.47)
\end{aligned}$$

The upper bound of I_1 can be derived as

$$\begin{aligned}
I_1&=\|y^k-y^*(\theta^k)\|^2-2\beta_k\mathbb{E}[\langle y^k-y^*(\theta^k),h_g^k\rangle|\mathcal{F}^k]+\beta_k^2\mathbb{E}[\|h_g^k\|^2|\mathcal{F}^k]\\
&\overset{(a)}{\leq}\|y^k-y^*(\theta^k)\|^2-2\beta_k\langle y^k-y^*(\theta^k),\nabla_y g(\theta^k,y^k)\rangle+\beta_k^2\|\nabla_y g(\theta^k,y^k)\|^2+\beta_k^2\sigma_{g_y}^2\\
&\overset{(b)}{\leq}\left(1-\frac{2\mu_g L_g}{\mu_g+L_g}\beta^k\right)\|y^k-y^*(\theta^k)\|^2+\beta_k\left(\beta_k-\frac{2}{\mu_g+L_g}\right)\|\nabla_y g(\theta^k,y^k)\|^2+\beta_k^2\sigma_{g_y}^2\\
&\overset{(c)}{\leq}\left(1-\frac{2\mu_g L_g}{\mu_g+L_g}\beta^k\right)\|y^k-y^*(\theta^k)\|^2+\beta_k^2\sigma_{g_y}^2 \qquad (3.48)
\end{aligned}$$

where (a) comes from the fact that $\mathrm{Var}[X]=\mathbb{E}[X^2]-\mathbb{E}[X]^2$, (b) follows from the μ_g-strong convexity and L_g smoothness of $g(\theta,y)$ [79, Theorem 2.1.11], and (c) follows from the choice of stepsize $\beta_k\leq\frac{\mu_g/L_g}{32(\mu_g+L_g)}\leq\frac{2}{\mu_g+L_g}$ in (3.27a).

The upper bound of I_2 can be derived as

$$\begin{aligned}
I_2&=\mathbb{E}\left[\left\|y^*(\theta^k)-y^*(\theta^{k+1})-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]\\
&\leq3\mathbb{E}\left[\left\|y^*(\theta^{k+1})-y^*(\theta^k)-\nabla_\theta y^*(\theta^k)(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]\\
&\quad+3\mathbb{E}\left[\left\|\left(\nabla_\theta y^*(\theta^k)-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\right)(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]\\
&\quad+3\mathbb{E}\left[\left\|\left(H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top\right)(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]. \qquad (3.49)
\end{aligned}$$

We first bound the first approximation error in the RHS of (3.49) by

$$
\begin{aligned}
&\left\|y^*(\theta^{k+1})-y^*(\theta^k)-\nabla_\theta y^*(\theta^k)(\theta^{k+1}-\theta^k)\right\|^2\\
=&\left\|\int_0^1 \nabla_\theta y^*(\theta^k+t(\theta^{k+1}-\theta^k))(\theta^{k+1}-\theta^k)dt-\nabla_\theta y^*(\theta^k)(\theta^{k+1}-\theta^k)\right\|^2\\
\leq&\int_0^1 \left\|\nabla_\theta y^*(\theta^k+t(\theta^{k+1}-\theta^k))-\nabla_\theta y^*(\theta^k)\right\|^2\|\theta^{k+1}-\theta^k\|^2dt\leq\frac{L_y^2}{2}\|\theta^{k+1}-\theta^k\|^4 \qquad (3.50)
\end{aligned}
$$

where the first inequality follows from the Cauchy-Schwarz inequality, and the second inequality follows from the L_y-Lipschitz continuity of $\nabla_\theta y^*(\theta)$ in Lemma 9.

Next we bound the second term in the RHS of (3.49) as

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\left(\nabla_\theta y^*(\theta^k)-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\right)(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]\\
\leq&\mathbb{E}\left[\left\|\nabla_\theta y^*(\theta^k)-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\right\|^2\left\|\theta^{k+1}-\theta^k\right\|^2|\mathcal{F}^k\right] \qquad (3.51)
\end{aligned}
$$

and likewise, the third term of (3.49) as

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\left(H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top\right)(\theta^{k+1}-\theta^k)\right\|^2|\mathcal{F}^k\right]\\
\leq&\mathbb{E}\left[\left\|H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top-(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top\right\|^2\left\|\theta^{k+1}-\theta^k\right\|^2|\mathcal{F}^k\right]. \qquad (3.52)
\end{aligned}
$$

We then bound the approximation error of $H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top$ in (3.51) by

$$
\begin{aligned}
&\left\|\nabla_\theta y^*(\theta^k)-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\right\|^2\\
=&\left\|H_{yy}\left(\theta^k,y^*(\theta^k)\right)^{-1}H_{\theta y}\left(\theta^k,y^*(\theta^k)\right)^\top-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\right\|^2\\
=&\Big\|H_{yy}\left(\theta^k,y^*(\theta^k)\right)^{-1}H_{\theta y}\left(\theta^k,y^*(\theta^k)\right)^\top-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}\left(\theta^k,y^*(\theta^k)\right)^\top\\
&\qquad+H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}\left(\theta^k,y^*(\theta^k)\right)^\top-H_{yy}(\theta^k,y^k)^{-1}H_{\theta y}(\theta^k,y^k)^\top\Big\|^2\\
\leq&2C_{g_{\theta y}}^2\left\|H_{yy}\left(\theta^k,y^*(\theta^k)\right)^{-1}-H_{yy}(\theta^k,y^k)^{-1}\right\|^2+\frac{2}{\mu_g^2}\left\|H_{\theta y}\left(\theta^k,y^*(\theta^k)\right)-H_{\theta y}(\theta^k,y^k)\right\|^2 \qquad (3.53)
\end{aligned}
$$

where the inequality follows from $\|H_{\theta y}(\theta,y)\|\leq C_{g_{\theta y}}$ and $H_{yy}(\theta,y)\succeq\mu_g I$.

Note that

$$
\begin{aligned}
&\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right)^{-1} - H_{yy}(\theta^k, y^k)^{-1}\right\|^2 \\
=&\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right)^{-1}\Big(H_{yy}\left(\theta^k, y^*(\theta^k)\right) - H_{yy}(\theta^k, y^k)\Big)H_{yy}(\theta^k, y^k)^{-1}\right\|^2 \\
\leq&\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right)^{-1}\right\|^2\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right) - H_{yy}(\theta^k, y^k)\right\|^2\left\|H_{yy}(\theta^k, y^k)^{-1}\right\|^2 \\
\leq&\frac{1}{\mu_g^4}\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right) - H_{yy}(\theta^k, y^k)\right\|^2
\end{aligned}
\tag{3.54}
$$

where the last inequality follows from $H_{yy}(\theta, y) \succeq \mu_g I$.

Therefore, we have

$$
\begin{aligned}
&\left\|\nabla_\theta y^*(\theta^k) - H_{yy}(\theta^k, y^k)^{-1} H_{\theta y}(\theta^k, y^k)^\top\right\|^2 \\
\leq&\frac{2C_{g_{\theta y}}^2}{\mu_g^4}\left\|H_{yy}\left(\theta^k, y^*(\theta^k)\right) - H_{yy}(\theta^k, y^k)\right\|^2 + \frac{2}{\mu_g^2}\left\|H_{\theta y}\left(\theta^k, y^*(\theta^k)\right) - H_{\theta y}(\theta^k, y^k)\right\|^2.
\end{aligned}
\tag{3.55}
$$

Following the steps towards (3.55), we bound the error of $(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top$ in (3.52) by

$$
\begin{aligned}
&\left\|(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top - H_{yy}(\theta^k, y^k)^{-1} H_{\theta y}(\theta^k, y^k)^\top\right\|^2 \\
=&\left\|(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top - H_{yy}(\theta^k, y^k)^{-1}(H_{\theta y}^k)^\top + H_{yy}(\theta^k, y^k)^{-1}(H_{\theta y}^k)^\top - H_{yy}(\theta^k, y^k)^{-1} H_{\theta y}(\theta^k, y^k)^\top\right\|^2 \\
\leq&2\left\|(H_{yy}^k)^{-1}(H_{\theta y}^k)^\top - H_{yy}(\theta^k, y^k)^{-1}(H_{\theta y}^k)^\top\right\|^2 \\
&\quad + 2\left\|H_{yy}(\theta^k, y^k)^{-1}(H_{\theta y}^k)^\top - H_{yy}(\theta^k, y^k)^{-1} H_{\theta y}(\theta^k, y^k)^\top\right\|^2 \\
\leq&\frac{2C_{g_{\theta y}}^2}{\mu_g^4}\left\|H_{yy}^k - H_{yy}(\theta^k, y^k)\right\|^2 + \frac{2}{\mu_g^2}\left\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\right\|^2
\end{aligned}
\tag{3.56}
$$

where the second inequality follows from $\|H_{\theta y}^k\| \leq C_{g_{\theta y}}$ and $H_{yy}^k \succeq \mu_g I$.

Plugging (3.50)-(3.56) back to (3.49), we have

$$
\begin{aligned}
I_2 \leq\ & \frac{3L_y^2}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^4|\mathcal{F}^k] + \frac{6C_{g_{\theta y}}^2}{\mu_g^4}\|H_{yy}(\theta^k, y^*(\theta^k)) - H_{yy}(\theta^k, y^k)\|^2\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k] \\
&+ \frac{6}{\mu_g^2}\|H_{\theta y}(\theta^k, y^*(\theta^k)) - H_{\theta y}(\theta^k, y^k)\|^2\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k] \\
&+ \frac{6C_{g_{\theta y}}^2}{\mu_g^4}\mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k] \\
&+ \frac{6}{\mu_g^2}\mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2\|\theta^{k+1} - \theta^k\|^2|\mathcal{F}^k].
\end{aligned}
\tag{3.57}
$$

Using the Lipschitz continuity of $H_{\theta y}(\theta, y)$ and $H_{yy}(\theta, y)$ in Assumption 1, from (3.57), we have

$$\begin{aligned}
I_2 \leq\ & \frac{3L_y^2}{2}\mathbb{E}[\|\theta^{k+1}-\theta^k\|^4|\mathcal{F}^k] + \frac{6}{\mu_g^2}\left(\frac{C_{g_{\theta y}}^2 L_{g_{yy}}}{\mu_g^2} + L_{g_{\theta y}}\right)\|y^k - y^*(\theta^k)\|^2\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k] \\
& + \frac{6C_{g_{\theta y}}^2}{\mu_g^4}\mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k] \\
& + \frac{6}{\mu_g^2}\mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2\|\theta^{k+1}-\theta^k\|^2|\mathcal{F}^k]. \qquad (3.58)
\end{aligned}$$

For any $p = 2, 4$, we next analyze quantity $\mathbb{E}[\|\theta^{k+1}-\theta^k\|^p|\mathcal{F}^k]$ in (3.58). Recall the simplified update (3.43). Therefore, we have $\|\theta^{k+1}-\theta^k\| = \alpha_k\|\bar{h}_f^k\|$ and

$$\begin{aligned}
\|\bar{h}_f^k\| &= \left\|\nabla_\theta f\left(\theta^k, y^k; \xi^k\right) - (H_{yy}^k)^{-1}H_{\theta y}^k\nabla_y f\left(\theta^k, y^k; \xi^k\right)\right\| \\
&\leq \left\|\nabla_\theta f(\theta^k, y^k; \xi^k)\right\| + \left\|(H_{yy}^k)^{-1}H_{\theta y}^k\nabla_y f\left(\theta^k, y^k; \xi^k\right)\right\| \\
&\overset{(a)}{\leq} \left\|\nabla_\theta f(\theta^k, y^k; \xi^k)\right\| + \frac{C_{g_{\theta y}}}{\mu_g}\left\|\nabla_y f(\theta^k, y^k; \xi^k)\right\| \qquad (3.59)
\end{aligned}$$

where (a) follows from the upper and lower projections of $H_{\theta y}^k$ and H_{yy}^k in (3.12).

Therefore, for $p = 2, 4$, we have

$$\begin{aligned}
\mathbb{E}[\|\bar{h}_f^k\|^p|\mathcal{F}^k, H_{\theta,y}^k, H_{yy}^k] &\leq 2^{p-1}\mathbb{E}\Big[\|\nabla_\theta f(\theta^k, y^k; \xi^k)\|^p|\mathcal{F}^k, H_{\theta,y}^k, H_{yy}^k\Big] \\
&\quad + 2^{p-1}\left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^p\mathbb{E}\Big[\|\nabla_y f(\theta^k, y^k; \xi^k)\|^p|\mathcal{F}^k, H_{\theta,y}^k, H_{yy}^k\Big] \\
&\leq 2^{p-1}\left(C_{f_\theta}^p + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^p C_{f_y}^p\right) \qquad (3.60)
\end{aligned}$$

where the last inequality from Assumption 3. And thus

$$\mathbb{E}\left[\|\theta^{k+1}-\theta^k\|^p|\mathcal{F}^k, H_{\theta y}^k, H_{yy}^k\right] \leq 2^{p-1}\left(C_{f_\theta}^p + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^p C_{f_y}^p\right)\alpha_k^p. \qquad (3.61)$$

Plugging (3.61) into (3.58), we have

$$\begin{aligned}
I_2 \leq\ & 12L_y^2 \left(C_{f_\theta}^4 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^4 C_{f_y}^4 \right) \alpha_k^4 \\
&+ \frac{12}{\mu_g^2} \left(\frac{C_{g_{\theta y}}^2 L_{g_{yy}}}{\mu_g^2} + L_{g_{\theta y}} \right) \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right) \|y^k - y^*(\theta^k)\|^2 \alpha_k^2 \\
&+ \frac{12 C_{g_{\theta y}}^2}{\mu_g^4} \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right) \mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2 | \mathcal{F}^k] \alpha_k^2 \\
&+ \frac{12}{\mu_g^2} \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right) \mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2 | \mathcal{F}^k] \alpha_k^2. \qquad (3.62)
\end{aligned}$$

Now let us define the constants as

$$\begin{aligned}
\tilde{c}_1 := \max \Bigg\{ & 12L_y^2 \left(C_{f_\theta}^4 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^4 C_{f_y}^4 \right), \frac{12}{\mu_g^2} \left(\frac{C_{g_{\theta y}}^2 L_{g_{yy}}}{\mu_g^2} + L_{g_{\theta y}} \right) \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right), \\
& \frac{12 C_{g_{\theta y}}^2}{\mu_g^4} \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right), \frac{12}{\mu_g^2} \left(C_{f_\theta}^2 + \left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2 C_{f_y}^2 \right) \Bigg\} \\
\tilde{c}_2 :=\ & \frac{2}{\mu_g + L_g} + \frac{\mu_g + L_g}{\mu_g L_g}, \qquad c := \tilde{c}_1 \tilde{c}_2.
\end{aligned}$$

Plugging the upper bounds of I_1 in (3.48) and I_2 in (3.62) into (3.47) with $\epsilon = \frac{\mu_g L_g}{\mu_g + L_g} \beta^k$, we have

$$\begin{aligned}
& \mathbb{E}\left[\|y^{k+1} - y^*(\theta^{k+1})\|^2 | \mathcal{F}^k\right] \\
\leq & \left(1 - \frac{\mu_g L_g}{\mu_g + L_g} \beta^k\right) \|y^k - y^*(\theta^k)\|^2 + \left(1 + \frac{\mu_g L_g}{\mu_g + L_g} \beta^k\right) \beta_k^2 \sigma_{g_y}^2 + \tilde{c}_1 \tilde{c}_2 \frac{\alpha_k^4}{\beta_k} \\
& + \tilde{c}_1 \tilde{c}_2 \frac{\alpha_k^2}{\beta_k} \|y^k - y^*(\theta^k)\|^2 + \tilde{c}_1 \tilde{c}_2 \mathbb{E}\left[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2 | \mathcal{F}^k\right] \frac{\alpha_k^2}{\beta_k} \\
& + \tilde{c}_1 \tilde{c}_2 \mathbb{E}\left[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2 | \mathcal{F}^k\right] \frac{\alpha_k^2}{\beta_k} \qquad (3.63)
\end{aligned}$$

where we have used the fact that

$$\begin{aligned}
\left(1 + \frac{\mu_g L_g}{\mu_g + L_g} \beta^k\right) \left(1 - \frac{2\mu_g L_g}{\mu_g + L_g} \beta^k\right) &\leq 1 - \frac{\mu_g L_g}{\mu_g + L_g} \beta^k \\
\left(1 + \left(\frac{\mu_g L_g}{\mu_g + L_g} \beta^k\right)^{-1}\right) &\leq \frac{1}{\beta_k} \left(\frac{2}{\mu_g + L_g} + \frac{\mu_g + L_g}{\mu_g L_g} \right) = \frac{\tilde{c}_2}{\beta_k}
\end{aligned}$$

where the last inequality uses $\beta_k \leq \frac{2}{\mu_g+L_g}$ in (3.27a). The proof is complete by defining $c := \tilde{c}_1\tilde{c}_2$.

3.4.5 Proof of Lemma 8

Proof: Recall that $g(\theta, y) = \mathbb{E}_\phi[g(\theta, y, \phi)]$. We only have access to the stochastic estimates of $\nabla^2_{\theta y} g\left(\theta, y\right), \nabla^2_{yy} g\left(\theta, y\right)$, that is

$$h^k_{yy}(\phi) := \nabla^2_{yy} g\left(\theta^k, y^k; \phi\right), \qquad h^k_{\theta y}(\phi) := \nabla^2_{\theta y} g\left(\theta^k, y^k; \phi\right). \tag{3.64}$$

For notational brevity in the analysis, we define

$$H_{\theta y}(\theta, y) := \nabla^2_{\theta y} g\left(\theta, y\right), \qquad\qquad H_{yy}(\theta, y) := \nabla^2_{yy} g\left(\theta, y\right). \tag{3.65}$$

and rewrite the update of (3.12) as

$$H^k_{\theta y} := \mathcal{P}_{\{X:\|X\|\leq C_{g_{\theta y}}\}}\left\{\hat{H}^k_{\theta y}\right\} \quad \text{with} \quad \hat{H}^k_{\theta y} := (1-\tau_k)(H^{k-1}_{\theta y} - h^{k-1}_{\theta y}(\phi^k)) + h^k_{\theta y}(\phi^k) \tag{3.66a}$$

$$H^k_{yy} := \mathcal{P}_{\{X:X\succeq\mu_g I\}}\left\{\hat{H}^k_{yy}\right\} \quad \text{with} \quad \hat{H}^k_{yy} := (1-\tau_k)\big(H^{k-1}_{yy} - h^{k-1}_{yy}(\phi^k)\big) + h^k_{yy}(\phi^k). \tag{3.66b}$$

To analyze the approximation error of $H^k_{\theta y}$, we decompose it into

$$\begin{aligned}\mathbb{E}\Big[\|H^k_{\theta y} - H_{\theta y}(\theta^k, y^k)\|^2\big|\mathcal{F}^k\Big] &\leq \mathbb{E}\left[\|\hat{H}^k_{\theta y} - H_{\theta y}(\theta^k, y^k)\|^2\big|\mathcal{F}^k\right] \\ &= \left\|\mathbb{E}\left[\hat{H}^k_{\theta y} - H_{\theta y}(\theta^k, y^k)|\mathcal{F}^k\right]\right\|^2 + \sum_{i,j}\mathrm{Var}\left[(\hat{H}^k_{\theta y} - H_{\theta y}(\theta^k, y^k))_{i,j}|\mathcal{F}^k\right]\end{aligned} \tag{3.67}$$

where the inequality holds since the projection onto the convex set $\{X : X \succeq \mu_g I\}$ is non-expansive, and the equality comes from the bias and variance decomposition that $\mathrm{Var}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$ for any random variable X.

We first analyze the bias term in (3.67) by

$$\begin{aligned}&\mathbb{E}\left[\hat{H}^k_{\theta y} - H_{\theta y}(\theta^k, y^k)|\mathcal{F}^k\right] \\ \overset{(3.12)}{=}\;& \mathbb{E}\left[(1-\tau_k)\left(H^{k-1}_{\theta y} + h^k_{\theta y}(\phi^k) - h^{k-1}_{\theta y}(\phi^k)\right) + \tau_k h^k_{\theta y}(\phi^k) - H_{\theta y}(\theta^k, y^k)|\mathcal{F}^k\right] \\ =\;& (1-\tau_k)\left(H^{k-1}_{\theta y} + H_{\theta y}(\theta^k, y^k) - H_{\theta y}(\theta^{k-1}, y^{k-1})\right) + \tau_k H_{\theta y}(\theta^k, y^k) - H_{\theta y}(\theta^k, y^k) \\ =\;& (1-\tau_k)\left(H^{k-1}_{\theta y} - H_{\theta y}(\theta^{k-1}, y^{k-1})\right).\end{aligned} \tag{3.68}$$

The variance term in (3.67) follows

$$
\begin{aligned}
&\sum_{i,j} \mathrm{Var}\left[(\hat{H}^k_{\theta y} - H_{\theta y}(\theta^k, y^k))_{i,j}|\mathcal{F}^k\right] = \sum_{i,j} \mathrm{Var}\left[(\hat{H}^k_{\theta y})_{i,j}|\mathcal{F}^k\right] \\
&\overset{(3.66a)}{=} \sum_{i,j} \mathrm{Var}\left[(1-\tau_k)(h^k_{\theta y}(\phi^k) - h^{k-1}_{\theta y}(\phi^k))_{i,j} + \tau_k(h^k_{\theta y}(\phi^k))_{i,j}|\mathcal{F}^k\right] \\
&\leq 2(1-\tau_k)^2 \sum_{i,j} \mathrm{Var}\left[(h^k_{\theta y}(\phi^k) - h^{k-1}_{\theta y}(\phi^k))_{i,j}|\mathcal{F}^k\right] + 2\tau_k^2 \sum_{i,j} \mathrm{Var}\left[(h^k_{\theta y}(\phi^k))_{i,j}|\mathcal{F}^k\right] \\
&\overset{(a)}{\leq} 2(1-\tau_k)^2 \mathbb{E}\left[\|h^k_{\theta y}(\phi^k) - h^{k-1}_{\theta y}(\phi^k)\|^2|\mathcal{F}^k\right] + 2\tau_k^2 \sum_{i,j} \mathrm{Var}\left[(h^k_{\theta y}(\phi^k))_{i,j}|\mathcal{F}^k\right] \\
&\overset{(b)}{\leq} 2(1-\tau_k)^2 \left(\bar{L}^2_{g_{\theta y}} + L^2_{g_{\theta y}}\right)\left(\|\theta^k - \theta^{k-1}\|^2 + \|y^k - y^{k-1}\|^2\right) + 2\tau_k^2\sigma^2_{g_{\theta y}} \qquad (3.69)
\end{aligned}
$$

where (a) uses $\mathrm{Var}[X] \leq \mathbb{E}[X]^2$ and (b) follows from Assumptions 1 and 3.

Therefore, plugging (3.68) and (3.69) into (3.67), we have

$$
\begin{aligned}
\mathbb{E}[\|H^k_{\theta y} - H_{\theta y}(\theta^k, y^k)\|^2|\mathcal{F}^k] &\leq (1-\tau_k)^2 \left\|H^{k-1}_{\theta y} - H_{\theta y}(\theta^{k-1}, y^{k-1})\right\|^2 + 2\tau_k^2\sigma^2_{g_{\theta y}} \\
&\quad + 2(1-\tau_k)^2\left(\bar{L}^2_{g_{\theta y}} + L^2_{g_{\theta y}}\right)\left(\|\theta^k - \theta^{k-1}\|^2 + \|y^k - y^{k-1}\|^2\right).
\end{aligned}
$$

Similarly, we can derive the approximation error of H^k_{yy} as

$$
\begin{aligned}
\mathbb{E}[\|H^k_{yy} - H_{yy}(\theta^k, y^k)\|^2|\mathcal{F}^k] &\leq (1-\tau_k)^2 \|H^{k-1}_{yy} - H_{yy}(\theta^{k-1}, y^{k-1})\|^2 + 2\tau_k^2\sigma^2_{g_{yy}} \\
&\quad + 2(1-\tau_k)^2\left(\bar{L}^2_{g_{yy}} + L^2_{g_{yy}}\right)\left(\|\theta^k - \theta^{k-1}\|^2 + \|y^k - y^{k-1}\|^2\right).
\end{aligned}
$$

The proof is then complete.

3.4.6 Proof of Theorem 2

Proof: Using Lemmas 6-8, we, respectively, bound the four difference terms in (3.33) and obtain

$$\begin{aligned}\mathbb{E}[\mathbb{V}^{k+1}] - \mathbb{E}[\mathbb{V}^k] \leq & -\frac{\alpha_k}{2}\mathbb{E}[\|\nabla f(\theta^k)\|^2] - \left(\frac{\mu_g L_g}{\mu_g + L_g}\beta_k - \tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k} - \alpha_k L_f^2\right)\mathbb{E}[\|y^k - y^*(\theta^k)\|^2] \\ & - \left(\tau_{k+1} - \tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k} - \frac{2\alpha_k C_{g_{\theta y}}^2 C_{f_y}^2}{\mu_g^4}\right)\mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2] \\ & - \left(\tau_{k+1} - \tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k} - \frac{2\alpha_k C_{f_y}^2}{\mu_g^2}\right)\mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2] \\ & + \frac{L_F}{2}\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2] + \left(1 + \frac{\mu_g L_g}{\mu_g + L_g}\beta^k\right)\beta_k^2\sigma_{g_y}^2 + \tilde{c}_1\tilde{c}_2\frac{\alpha_k^4}{\beta_k} + 4\tau_{k+1}^2\sigma_{g_y}^2 \\ & + 2(1-\tau_{k+1})^2\tilde{c}_3\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2] + 2(1-\tau_{k+1})^2\tilde{c}_3\mathbb{E}[\|y^{k+1} - y^k\|^2]. \quad (3.70)\end{aligned}$$

where the constant is defined as $\tilde{c}_3 := \bar{L}_{g_{\theta y}}^2 + L_{g_{\theta y}}^2 + \bar{L}_{g_{yy}}^2 + L_{g_{yy}}^2$.

Note that using the y-update (3.11b), we also have

$$\begin{aligned}\mathbb{E}[\|y^{k+1} - y^k\|^2] &= \mathbb{E}\left[\left\|\beta_k h_g^k - (H_{yy}^k)^{-1}H_{\theta y}^k(\theta^{k+1} - \theta^k)\right\|^2\right] \\ &\leq 2\beta_k^2\mathbb{E}\left[\|h_g^k\|^2\right] + 2\mathbb{E}\left[\|(H_{yy}^k)^{-1}\|^2\|H_{\theta y}^k\|^2\left\|\theta^{k+1} - \theta^k\right\|^2\right] \\ &\overset{(a)}{\leq} 2\beta_k^2\mathbb{E}\left[\|\nabla_y g(\theta^k, y^k)\|^2\right] + 2\beta_k^2\sigma_{g_y}^2 + 2\mathbb{E}\left[\|(H_{yy}^k)^{-1}\|^2\|H_{\theta y}^k\|^2\left\|\theta^{k+1} - \theta^k\right\|^2\right] \\ &\overset{(b)}{\leq} 2\beta_k^2\mathbb{E}\left[\|\nabla_y g(\theta^k, y^k)\|^2\right] + 2\beta_k^2\sigma_{g_y}^2 + 2\left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2\mathbb{E}\left[\left\|\theta^{k+1} - \theta^k\right\|^2\right] \\ &\overset{(c)}{\leq} 2\beta_k^2 L_g^2\mathbb{E}[\|y^k - y^*(\theta^k)\|^2] + 2\beta_k^2\sigma_{g_y}^2 + 2\left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2\mathbb{E}[\|\theta^{k+1} - \theta^k\|^2] \quad (3.71)\end{aligned}$$

where (a) follows from $\mathbb{E}[X^2] = \mathrm{Var}[X] + \mathbb{E}[X]^2$ and Assumption 3, (b) uses the upper and lower projections of $H_{\theta y}^k$ and H_{yy}^k in (3.12), and (c) is due to $\nabla_y g(\theta^k, y^*(\theta^k)) = 0$ as well as Assumption 1.

Selecting parameter $\tau_k = \frac{1}{\sqrt{K}}$ and using (3.70)-(3.71), we have

$$\begin{aligned}\mathbb{E}[\mathbb{V}^{k+1}]-\mathbb{E}[\mathbb{V}^{k}] \leq & -\frac{\alpha_k}{2}\mathbb{E}[\|\nabla f(\theta^k)\|^2]+\left(\frac{L_F}{2}+2\tilde{c}_3+4\tilde{c}_3\left(\frac{C_{g_{\theta y}}}{\mu_g}\right)^2\right)\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2]\\
& -\left(\frac{\mu_g L_g}{\mu_g+L_g}\beta_k-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\alpha_k L_f^2-8\beta_k^2L_g^2\right)\mathbb{E}[\|y^k-y^*(\theta^k)\|^2]\\
& -\left(\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2C_{g_{\theta y}}^2C_{f_y}^2\alpha_k}{\mu_g^4}\right)\mathbb{E}[\|H_{yy}^k-H_{yy}(\theta^k,y^k)\|^2]\\
& -\left(\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2C_{f_y}^2\alpha_k}{\mu_g^2}\right)\mathbb{E}[\|H_{\theta y}^k-H_{\theta y}(\theta^k,y^k)\|^2]\\
& +\left(1+\frac{\mu_g L_g}{\mu_g+L_g}\beta^k\right)\beta_k^2\sigma_{g_y}^2+\tilde{c}_1\tilde{c}_2\frac{\alpha_k^4}{\beta_k}+\frac{4\sigma_{g_y}^2}{K}+8\beta_k^2L_g^2\sigma_{g_y}^2. \end{aligned} \tag{3.72}$$

Choosing the stepsize α_k as (3.27), it will lead to (cf. $c := \tilde{c}_1\tilde{c}_2$)

$$\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2C_{g_{\theta y}}^2C_{f_y}^2\alpha_k}{\mu_g^4}\overset{(a)}{\geq}\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\alpha_k-\frac{2C_{g_{\theta y}}^2C_{f_y}^2\alpha_k}{\mu_g^4}\overset{(b)}{\geq}0 \tag{3.73a}$$

$$\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2C_{f_y}^2\alpha_k}{\mu_g^2}\overset{(c)}{\geq}\frac{1}{\sqrt{K}}-\tilde{c}_1\tilde{c}_2\alpha_k-\frac{2C_{f_y}^2\alpha_k}{\mu_g^2}\overset{(d)}{\geq}0 \tag{3.73b}$$

where both (a) and (c) follow from $\alpha_k \leq \beta_k$ in (3.27b); and (b) and (d) follow from the second and the third terms in (3.27b). In addition, choosing the stepsize β_k as (3.27) will lead to

$$\begin{aligned}\frac{\mu_g L_g}{\mu_g+L_g}\beta_k-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\alpha_k L_f^2-8\beta_k^2L_g^2 & \overset{(e)}{\geq}\frac{\mu_g L_g}{\mu_g+L_g}\beta_k-(\tilde{c}_1\tilde{c}_2+L_f^2)\alpha_k-8\beta_k^2L_g^2\\
& \overset{(f)}{\geq}\frac{\mu_g L_g\beta_k}{2(\mu_g+L_g)}-8\beta_k^2L_g^2\overset{(g)}{\geq}\frac{\mu_g L_g\beta_k}{4(\mu_g+L_g)}\end{aligned} \tag{3.73c}$$

where (e) follows from $\alpha_k \leq \beta_k$ in (3.27b), (f) is due to the last terms in (3.27b), and (g) uses (3.27a).

Using (3.73) to cancel terms in (3.72) and using (3.61) to bound $\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2]$, we are able to get

$$\mathbb{E}[\mathbb{V}^{k+1}]-\mathbb{E}[\mathbb{V}^{k}]\leq-\frac{\mu_g L_g\beta_k}{4(\mu_g+L_g)}\mathbb{E}[\|y^k-y^*(\theta^k)\|^2]-\frac{\alpha_k}{2}\mathbb{E}[\|\nabla f(\theta^k)\|^2]+\mathcal{O}\left(\frac{1}{K}\right) \tag{3.74}$$

from which we can reach Theorem 2 after telescoping the both sides of (3.74).

3.4.7 Proof of Theorem 3

For the strong-convex case, we slightly modify the update of θ^k to

$$\theta^{k+1} = \mathcal{P}_\Theta\left(\theta^k - \alpha_k \bar{h}_f^k\right) \tag{3.75}$$

where $\bar{h}_f^k$ is defined as (3.43) and $\mathcal{P}_\Theta$ denotes the projection on set Θ.

Slightly different from the Lyapunov function (3.32), we define the following Lyapunov function

$$\mathbb{V}^k := \|\theta^k - \theta^*\|^2 + \|y^k - y^*(\theta^k)\|^2 + \|H_{yy}^k - \nabla_{yy}^2 g(\theta^k, y^k)\|^2 + \|H_{\theta y}^k - \nabla_{\theta y}^2 g(\theta^k, y^k)\|^2.$$

Lemma 10 *Suppose Assumptions 1–3 hold and $F(\theta)$ is μ-strongly convex. Then θ^k satisfies*

$$\begin{aligned}\mathbb{E}[\|\theta^{k+1} - \theta^*\|^2] \leq (1-\mu\alpha_k)\mathbb{E}[\|\theta^k - \theta^*\|^2] + \frac{2L_f^2}{\mu}\alpha_k\mathbb{E}[\|y^k - y^*(\theta^k)\|^2] + \alpha_k^2\mathbb{E}[\|\bar{h}_f^k\|^2] \\ + \frac{4C_{g_{\theta y}}^2 C_{f_y}^2}{\mu_g^4\mu}\alpha_k\mathbb{E}[\|H_{yy}^k - H_{yy}(\theta^k, y^k)\|^2] + \frac{4C_{f_y}^2}{\mu_g^2\mu}\alpha_k\mathbb{E}[\|H_{\theta y}^k - H_{\theta y}(\theta^k, y^k)\|^2]\end{aligned} \tag{3.76}$$

where L_f, L_F are defined in Lemma 9, and $C_{g_{\theta y}}$ is the projection radius of $H_{\theta y}^k$ in (3.12a).

Proof: We start with

$$\begin{aligned}\mathbb{E}[\|\theta^{k+1} - \theta^*\|^2|\mathcal{F}^k] &\overset{(a)}{\leq} \mathbb{E}[\|\theta^k - \alpha_k\bar{h}_f^k - \theta^*\|^2|\mathcal{F}^k] \\ &= \|\theta^k - \theta^*\|^2 - 2\alpha_k\langle\theta^k - \theta^*, \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle + \alpha_k^2\mathbb{E}[\|\bar{h}_f^k\|^2|\mathcal{F}^k] \\ &= \|\theta^k - \theta^*\|^2 - 2\alpha_k\langle\theta^k - \theta^*, \nabla f(\theta^k)\rangle \\ &\quad + 2\alpha_k\langle\theta^k - \theta^*, \nabla f(\theta^k) - \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle + \alpha_k^2\mathbb{E}[\|\bar{h}_f^k\|^2|\mathcal{F}^k] \\ &\overset{(b)}{\leq} \|\theta^k - \theta^*\|^2 - 2\alpha_k\langle\theta^k - \theta^*, \nabla f(\theta^k) - \nabla f(\theta^*)\rangle \\ &\quad + 2\alpha_k\langle\theta^k - \theta^*, \nabla f(\theta^k) - \mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle + \alpha_k^2\mathbb{E}[\|\bar{h}_f^k\|^2|\mathcal{F}^k]\end{aligned} \tag{3.77}$$

where (a) follows the fact that $\mathcal{P}_\Theta$ is non-expansive, and (b) follows the optimality condition that $\langle\nabla f(\theta^*), \theta - \theta^*\rangle \geq 0$ for any $\theta \in \Theta$.

Using the μ-strong convexity of $F(\theta)$, it follows that

$$-\langle \theta^k-\theta^*, \nabla f(\theta^k)-\nabla f(\theta^*)\rangle \leq -\mu\|\theta^k-\theta^*\|^2 \tag{3.78}$$

plugging which into (3.77) leads to

$$\begin{aligned}\mathbb{E}\left[\|\theta^{k+1}-\theta^*\|^2|\mathcal{F}^k\right] &\leq (1-2\mu\alpha_k)\|\theta^k-\theta^*\|^2+2\alpha_k\langle\theta^k-\theta^*,\nabla f(\theta^k)-\mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\rangle+\alpha_k^2\mathbb{E}\left[\|\bar{h}_f^k\|^2|\mathcal{F}^k\right]\\ &\overset{(c)}{\leq}(1-\mu\alpha_k)\|\theta^k-\theta^*\|^2+\frac{\alpha_k}{\mu}\left\|\nabla f(\theta^k)-\mathbb{E}[\bar{h}_f^k|\mathcal{F}^k]\right\|^2+\alpha_k^2\mathbb{E}\left[\|\bar{h}_f^k\|^2|\mathcal{F}^k\right]\end{aligned}$$

where (c) uses the Young's inequality. Plugging (3.45) into the above completes the proof.

Similar to (3.33), we first quantify the difference between consecutive Lyapunov functions as

$$\begin{aligned}\mathbb{V}^{k+1}-\mathbb{V}^k=&\underbrace{\|\theta^{k+1}-\theta^*\|^2-\|\theta^k-\theta^*\|^2}_{\text{Lemma 10}}+\underbrace{\|y^{k+1}-y^*(\theta^{k+1})\|^2-\|y^k-y^*(\theta^k)\|^2}_{\text{Lemma 7}}\\ &+\underbrace{\|H_{yy}^{k+1}-\nabla_{yy}^2 g(\theta^{k+1},y^{k+1})\|^2-\|H_{yy}^k-\nabla_{yy}^2 g(\theta^k,y^k)\|^2}_{\text{Lemma 8}}\\ &+\underbrace{\|H_{\theta y}^{k+1}-\nabla_{\theta y}^2 g(\theta^{k+1},y^{k+1})\|^2-\|H_{\theta y}^k-\nabla_{\theta y}^2 g(\theta^k,y^k)\|^2}_{\text{Lemma 8}}.\end{aligned} \tag{3.79}$$

Using Lemmas 7-8 and 10 and defining $\tilde{c}_3:=\bar{L}_{g_{\theta y}}^2+L_{g_{\theta y}}^2+\bar{L}_{g_{yy}}^2+L_{g_{yy}}^2$, we obtain

$$\begin{aligned}\mathbb{E}[\mathbb{V}^{k+1}]-\mathbb{E}[\mathbb{V}^k]\leq&-\mu\alpha_k\mathbb{E}[\|\theta^k-\theta^*\|^2]-\left(\frac{\mu_g L_g\beta_k}{\mu_g+L_g}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2L_f^2\alpha_k}{\mu}\right)\mathbb{E}[\|y^k-y^*(\theta^k)\|^2]\\ &-\left(\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{g_{\theta y}}^2C_{f_y}^2}{\mu_g^4\mu}\alpha_k\right)\mathbb{E}[\|H_{yy}^k-H_{yy}(\theta^k,y^k)\|^2]\\ &-\left(\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{f_y}^2}{\mu_g^2\mu}\alpha_k\right)\mathbb{E}[\|H_{\theta y}^k-H_{\theta y}(\theta^k,y^k)\|^2]\\ &+\alpha_k^2\mathbb{E}[\|\bar{h}_f^k\|^2]+\left(1+\frac{\mu_g L_g}{\mu_g+L_g}\beta^k\right)\beta_k^2\sigma_{g_y}^2+\tilde{c}_1\tilde{c}_2\frac{\alpha_k^4}{\beta_k}+4\tau_{k+1}^2\sigma_{g_y}^2\\ &+2(1-\tau_{k+1})^2\tilde{c}_3\mathbb{E}[\|\theta^{k+1}-\theta^k\|^2]+2(1-\tau_{k+1})^2\tilde{c}_3\mathbb{E}[\|y^{k+1}-y^k\|^2].\end{aligned} \tag{3.80}$$

Note that for the projected update (3.75), (3.60) still holds. Plugging (3.71) and (3.60) into (3.80), we have

$$\begin{aligned}\mathbb{E}[\mathbb{V}^{k+1}]-\mathbb{E}[\mathbb{V}^k]\leq{}&-\mu\alpha_k\mathbb{E}[\|\theta^k-\theta^*\|^2]+\underbrace{\left(2+4\tilde{c}_3+8\tilde{c}_3\Big(\frac{C_{g_{\theta y}}}{\mu_g^2}\Big)^2\right)\left(C_{f_\theta}^2+\Big(\frac{C_{g_{\theta y}}}{\mu_g}\Big)^2C_{f_y}^2\right)}_{\tilde{c}_4:=}\alpha_k^2\\&-\left(\frac{\mu_gL_g\beta_k}{\mu_g+L_g}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2L_f^2\alpha_k}{\mu}-8\beta_k^2L^2L_g^2\right)\mathbb{E}[\|y^k-y^*(\theta^k)\|^2]\\&-\left(\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{g_{\theta y}}^2C_{f_y}^2}{\mu_g^4\mu}\alpha_k\right)\mathbb{E}[\|H_{yy}^k-H_{yy}(\theta^k,y^k)\|^2]\\&-\left(\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{f_y}^2}{\mu_g^2\mu}\alpha_k\right)\mathbb{E}[\|H_{\theta y}^k-H_{\theta y}(\theta^k,y^k)\|^2]\\&+\left(1+\frac{\mu_gL_g}{\mu_g+L_g}\beta^k\right)\beta_k^2\sigma_{g_y}^2+\tilde{c}_1\tilde{c}_2\frac{\alpha_k^4}{\beta_k}+4\tau_{k+1}^2\sigma_{g_y}^2+8L_g^2\beta_k^2\sigma_{g_y}^2.\end{aligned}\tag{3.81}$$

We choose the stepsizes α_k,β_k,τ_k as (3.30) to guarantee that (cf. $c:=\tilde{c}_1\tilde{c}_2$)

$$\begin{aligned}&\text{(a)}\quad\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{g_{\theta y}}^2C_{f_y}^2}{\mu_g^4\mu}\alpha_k\geq\frac{\beta_k}{4};\quad\text{(b)}\quad\tau_{k+1}-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{4C_{f_y}^2}{\mu_g^2\mu}\alpha_k\geq\frac{\beta_k}{4}\\&\text{(c)}\quad\frac{\mu_gL_g}{\mu_g+L_g}\beta_k-\tilde{c}_1\tilde{c}_2\frac{\alpha_k^2}{\beta_k}-\frac{2L_f^2}{\mu}\alpha_k-8\beta_k^2L^2L_g^2\geq\frac{\mu_gL_g}{4(\mu_g+L_g)}.\end{aligned}\tag{3.82}$$

Therefore, plugging (3.82) into (3.81), we have

$$\begin{aligned}\mathbb{E}[\mathbb{V}^{k+1}]-\mathbb{E}[\mathbb{V}^k]&\leq-\mu\alpha_k\mathbb{E}\Big[\|\theta^k-\theta^*\|^2\Big]-\frac{\mu_gL_g}{4(\mu_g+L_g)}\beta_k\mathbb{E}\Big[\|y^k-y^*(\theta^k)\|^2\Big]\\&\quad-\frac{\beta_k}{4}\mathbb{E}\Big[\|H_{yy}^k-H_{yy}(\theta^k,y^k)\|^2\Big]-\frac{\beta_k}{4}\mathbb{E}\Big[\|H_{\theta y}^k-H_{\theta y}(\theta^k,y^k)\|^2\Big]+\tilde{c}_6\beta_k^2\\&\leq-\tilde{c}_5\beta_k\mathbb{E}[\mathbb{V}^k]+\tilde{c}_6\beta_k^2\end{aligned}\tag{3.83}$$

where the first and second inequalities hold since we define

$$\begin{aligned}\tilde{c}_5&:=\min\left\{\frac{\mu\alpha_k}{\beta_k},\frac{\mu_gL_g}{4(\mu_g+L_g)},\frac{1}{4}\right\}=\mathcal{O}(1)\\\tilde{c}_6&:=\left(1+\frac{\mu_gL_g}{\mu_g+L_g}\beta^k\right)\sigma_{g_y}^2+\frac{\alpha_k^2}{4\beta_k}+4\sigma_{g_y}^2+8L_g^2\sigma_{g_y}^2+\tilde{c}_4=\mathcal{O}(1).\end{aligned}\tag{3.84}$$

If we choose $\beta_k = \frac{2}{\tilde{c}_5(K_0+k)}$, where K_0 is a sufficiently large constant, then we have

$$\begin{aligned}
\mathbb{E}[\mathbb{V}^K] &\leq \prod_{k=0}^{K-1}(1-\tilde{c}_5\beta_k)\mathbb{V}^0 + \tilde{c}_6\sum_{k=0}^{K-1}\beta_k^2\prod_{j=k+1}^{k-1}(1-\tilde{c}_5\beta_j) \\
&\leq \frac{(K_0-2)(K_0-1)}{(K_0+K-2)(K_0+K-1)}\mathbb{V}^0 + \frac{\tilde{c}_6}{\tilde{c}_5^2}\sum_{k=0}^{K-1}\frac{4}{(k+K_0)^2}\frac{(k+K_0-1)(k+K_0)}{(K+K_0-2)(K+K_0-1)} \\
&\leq \frac{(K_0-1)^2}{(K_0+K-1)^2}\mathbb{V}^0 + \frac{4\tilde{c}_6 K}{\tilde{c}_5^2(K+K_0-1)^2} \qquad (3.85)
\end{aligned}$$

from which the proof is complete.

CHAPTER 4

Communication-Adaptive Stochastic Gradients for Distributed Learning

4.1 Introduction

Stochastic gradient descent (SGD) method [93] is prevalent in solving large-scale machine learning problems during the last decades. Although simple to use, the plain-vanilla SGD often becomes less efficient when it is applied to the distributed setting, especially in terms of the communication efficiency.

In this chapter, we aim to solve the distributed learning problem in a communication-efficient fashion while maintaining the learning accuracy. Consider a setting consisting of a cloud server and a set of M devices (workers) collected in $\mathcal{M} := \{1, \ldots, M\}$. Each device m has its local dataset $\{\xi_n,\, n \in \mathcal{N}_m\}$, which defines the loss function of device m as

$$\mathcal{L}_m(\theta) := \sum_{n \in \mathcal{N}_m} \ell(\theta; \xi_n), \quad m \in \mathcal{M} \tag{4.1}$$

where $\theta \in \mathbb{R}^p$ is the sought vector (e.g., parameters of a prediction model) and ξ_n is a data sample. For example, in linear regression, $\ell(\theta; \xi_n)$ is the square loss; and, in deep learning, $\ell(\theta; \xi_n)$ is the loss function of a neural network, and θ concatenates the weights. The goal is to solve

$$\min_{\theta \in \mathbb{R}^p} \ \mathcal{L}(\theta) \quad \text{with} \quad \mathcal{L}(\theta) := \frac{1}{M} \sum_{m \in \mathcal{M}} \mathcal{L}_m(\theta). \tag{4.2}$$

Problem (4.2) also arises in a number of areas, such as multi-agent optimization [77], distributed signal processing [34], and distributed machine learning [17]. While our algorithms can be applied to other settings, we focus on the setting that for bandwidth and privacy concerns, local data $\{\xi_n, n \in \mathcal{N}_m\}$ at each worker m are not uploaded to the server. This setting naturally arises in e.g., federated learning, in which collaboration is needed through communication between the server and multiple workers (e.g., mobile devices).

To solve (4.2), we can in principle apply the distributed version of SGD. In this case, at iteration k, the server broadcasts the current model θ^k to *all* the workers; each worker m computes $\nabla\ell(\theta^k; \xi_m^k)$ using a randomly selected sample or a minibatch of samples $\{\xi_m^k\} \subseteq \{\xi_n, n \in \mathcal{N}_m\}$, and then uploads it to the server; and once receiving stochastic gradients from all workers, the server updates the model parameters via

$$\textbf{SGD} \qquad \theta^{k+1} = \theta^k - \frac{\eta_k}{M} \sum_{m \in \mathcal{M}} \nabla\ell(\theta^k; \xi_m^k) \tag{4.3}$$

where $\eta_k > 0$ is the (possibly) time-varying stepsize used at iteration k. When $\nabla\ell(\theta^k; \xi_m^k)$ is an unbiased gradient estimator of $\mathcal{L}_m(\theta)$, the convergence of SGD update (4.3) is guaranteed [7]. To implement (4.3), however, the server has to communicate with *all* workers to obtain fresh $\{\nabla\ell(\theta^k; \xi_m^k)\}$. This prevents the efficient implementation of SGD in scenarios where communication between the server and the workers is costly [73]. For example, consider using SGD to iteratively train an image classification model over a group of wireless devices. The start-of-the-art deep neural network models (e.g., ResNet, LSTM) for computer vision, speech and natural language processing tasks involve millions of parameters (e.g., 500MB). This training process is costly because one SGD update generates around 500 MB data on each device's up- and down-link transmission, and SGD takes thousands of iterations to converge. Therefore, *our goal* is to find the parameter θ that minimizes (4.2) with minimal communication overhead.

4.1.1 Related work

Communication-efficient distributed learning methods have gained popularity recently [76, 48]. Most methods belong to two categories: c1) reducing the bits per communication round; and, c2) reducing the communication rounds.

Reducing communication bits. For c1), methods are centered around the ideas of *quantization* and *sparsification*.

Quantization has been successfully applied to wireless sensor networks [87, 75]. In the context of distributed machine learning, a 1-bit and multi-bits quantization methods have been developed in [98, 4, 2]. Other variants of quantized gradient schemes include error compensation [122], variance-reduced quantization [129], and quantization to a ternary vector [121].

Sparsification amounts to transmitting only gradient coordinates with large enough magnitudes exceeding a certain threshold [1]. To avoid losing information of skipping communication, small gradient components will be accumulated and then transmitted when they are large enough [66, 105, 3, 119, 114].

Quantization and sparsification address c1) but not address c2), so they are still affected by latencies due to initiating communication, queuing, and propagating messages [85].

Reducing communication rounds. Methods using periodic averaging include elastic averaging SGD [132], local SGD (FedAvg) [73, 106, 115, 126, 52, 36] and momentum SGD [116]. Except [50, 115, 36], local SGD methods follow a fixed communication schedule. They work well in the *homogeneous* setting where data are i.i.d. over all workers, but often sacrifice the learning accuracy in the non-i.i.d. case. Work tailored for the heterogeneous setting includes FedProx [60]. Other methods that reduce the number of iterations include the gradient tracking [108, 59], primal-dual update [69, 71], opportunistic communication [96], and higher-order methods [100, 133]. Roughly speaking, algorithms in [60, 42, 100, 133] reduce communication by increasing local gradient computation.

This chapter is based on the method of lazily aggregated gradient (LAG) [8, 109]. LAG is adaptive and works well for the *heterogeneous* setting. Parameters in LAG are updated at the server, and workers only upload information that is informative enough. LAG has great performance with full gradient, but its performance degrades significantly with stochastic gradients, which make its rule of communication highly unreliable.

4.1.2 Our approach

This chapter proposes **L**azily **A**ggregated **S**tochastic **G**radient (**LASG**), which includes a set of SGD-based methods that considerably reduce the communication of distributed SGD. Compared with popular communication-efficient algorithms such as local SGD [73, 106, 115, 126], our LASG does not sacrifice learning accuracy in the non-i.i.d. settings. Observing that not all communications between the server and the workers are equally important, LASG uses conditions to decide communication adaptively. When a worker skips a round of communication, the server uses its stale gradient to perform parameter updates.

Define $\mathcal{M}^k$ as the set of uploading workers at iteration k, and define τ_m^k as the staleness of the gradient from worker m used at iteration k. LASG has the following update

$$\theta^{k+1} = \theta^k - \frac{\eta_k}{M} \sum_{m \in \mathcal{M} \setminus \mathcal{M}^k} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \frac{\eta_k}{M} \sum_{m \in \mathcal{M}^k} \nabla \ell(\theta^k; \xi_m^k) \tag{4.4}$$

or equivalently (see also Figure 4.1)

$$\textbf{Generic LASG} \qquad \theta^{k+1} = \theta^k - \eta_k \nabla^k \tag{4.5}$$
$$\text{with} \quad \nabla^k = \nabla^{k-1} + \frac{1}{M} \sum_{m \in \mathcal{M}^k} \delta_m^k$$

where the stochastic gradient innovation is defined as

$$\delta_m^k := \nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}). \tag{4.6}$$

The staleness $\{\tau_m^k\}$ depend on $\mathcal{M}^k$: at iteration k, if worker $m \notin \mathcal{M}^k$, the server increases staleness $\tau_m^{k+1} = \tau_m^k + 1$; otherwise, worker m uploads its stochastic gradient, and the server resets $\tau_m^{k+1} = 1$.

Clearly, selection of subset $\mathcal{M}^k$ is critical. The challenges are 1) the importance of each communication round is dynamic, thus a fixed condition is ineffective; and 2) checking the

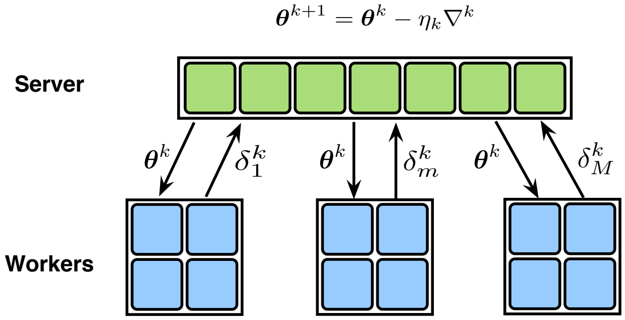

Figure 4.1: Generic LASG implementation.

condition must be numerically efficiently. To address these challenges, we develop two types of adaptive condition based on different *communication*, *computation* and *memory* requirements. The first type is computed by each worker (**WK**), and the second by the server (**PS**).

LASG-WK: At iteration k, the server broadcasts θ^k to all workers; each worker m computes $\nabla\ell(\theta^k;\xi_m^k)$, and checks whether $m \in \mathcal{M}^k$; only those in $\mathcal{M}^k$ upload δ_m^k to the server, which executes (4.5).

LASG-PS: At iteration k, the server determines $\mathcal{M}^k$ and sends θ^k to those workers $m \in \mathcal{M}^k$; each worker $m\in\mathcal{M}^k$ computes $\nabla\ell(\theta^k;\xi_m^k)$ and uploads δ_m^k; those workers $m\notin\mathcal{M}^k$ do nothing; the server executes (4.5);

How $\mathcal{M}^k$ are computed are deferred to Chapter 4.2. We summarize the contributions of this chapter as follows.

1) We introduce LASG, a set of communication-skipping methods for distributed SGD. It reuses stale stochastic gradients to reduce redundant communication.

2) We establish convergence of our proposed methods. The convergence rates match those of SGD.

3) We tested LASG on logistic regression and neural network training and confirm its performance gains.

Metric	**Communication**		**Computation**		**Memory**	
Algorithm	**PS→WK** m	**WK** m**→PS**	**PS**	**WK** m	**PS**	**WK** m
Sync SGD	always	always	(4.3)	(4.3)	$\mathcal{O}(p)$	/
LASG-WK1	always	if $m\in\mathcal{M}^k$	(4.5)	(4.9)	$\mathcal{O}(p)$	$\mathcal{O}(p)$
LASG-WK2	always	if $m\in\mathcal{M}^k$	(4.5)	(4.12)	$\mathcal{O}(p)$	$\mathcal{O}(p)$
LASG-PS	if $m\in\mathcal{M}^k$	if $m\in\mathcal{M}^k$	(4.5), (4.14)	if $m\in\mathcal{M}^k$	$\mathcal{O}(Mp)$	$\mathcal{O}(p)$
LASG-PSE	if $m\in\mathcal{M}^k$	if $m\in\mathcal{M}^k$	(4.5), (4.16)	if $m\in\mathcal{M}^k$	$\mathcal{O}(Mp)$	$\mathcal{O}(p)$

Table 4.1: A comparison of communication, computation and memory requirements. **PS** denotes the parameter server, **WK** denotes the worker, **PS→WK** m is the download from the server to worker m, and **WK** m → **PS** is the upload from worker m to the server.

4.1.3 Why LAG does not work well with SGD?

Let us revisit the LAG method [8] and provide why it works poorly with stochastic gradients.

Similar to what is described above, LAG has both WK and PS types of conditions to decide $\mathcal{M}^k$. Since they are equally ineffective with stochastic gradients, we limit our discussion to LAG-WK. Applying LAG-WK to stochastic gradients amounts to, in the condition of [8], replacing worker m's gradient by its stochastic gradient, that is, *exclude* m from $\mathcal{M}^k$ if

$$\left\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\right\|^2\leq\frac{c}{d_{\max}}\sum_{d=1}^{d_{\max}}\left\|\theta^{k+1-d}-\theta^{k-d}\right\|^2, \tag{4.7}$$

where $c\geq 0$ is a pre-defined constant, and $d_{\max}$ is the number of consecutive past iterates. This condition compares the new stochastic gradient to the stale copy at the server; if the difference is small compared to the recent changes in θ, then the server will reuse the stale copy.

When used with (standard) gradients, LAG [8] proves the condition leads to *"larger descent per upload"*. Unfortunately, the two stochastic gradients in (4.7) are evaluated with two different samples, ξ_m^k and $\xi_m^{k-\tau_m^k}$. The left-hand side (LHS) is almost never small. So, (4.7) becomes ineffective at judging the contribution of $\nabla\ell(\theta^k;\xi_m^k)$ to the *stochastic* descent.

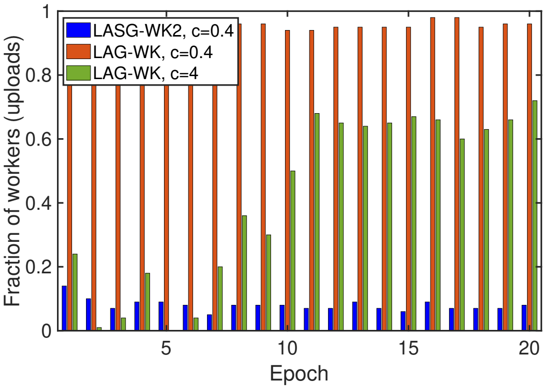

Figure 4.2: Comparison of upload numbers (10 iterations per epoch). Applying LAG-WK with stochastic gradients is ineffective. Even using an aggressive parameter $c = 4$, it is significantly less effective than LASG-WK2 (proposed).

Figure 4.2 compares the stochastic LAG and one of our new algorithms LASG-WK2 (introduced later) on a synthetically generated logistic regression task, which demonstrates that the stochastic LAG is ineffective in saving communication — when c is small (e.g., 0.4), (4.7) is almost never satisfied due to the inherent variance of the computed stochastic gradients; and when c is large (e.g., 4), (4.7) is satisfied only initially. Mathematically, this can be explained by expanding the LHS of (4.7) by (see the supplemental material for the deduction)

$$\mathbb{E}\left[\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\|^2\right] \tag{4.8a}$$

$$\geq \frac{1}{2}\mathbb{E}\Big[\Big\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\mathcal{L}_m(\theta^k)\Big\|^2\Big] \tag{4.8b}$$

$$+\frac{1}{2}\mathbb{E}\Big[\big[\Big\|\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})-\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})\Big\|^2\big]\Big] \tag{4.8c}$$

$$-\mathbb{E}[\|\nabla\mathcal{L}_m(\theta^k)-\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})\|^2]. \tag{4.8d}$$

When θ^k converges, e.g., $\theta^k \to \theta^*$, the right-hand side (RHS) of (4.7) $\left\|\theta^{k+1-d}-\theta^{k-d}\right\|^2 \to 0$. But the LHS of (4.7) does not since the gradient variances in (4.8b) and (4.8c) do not vanish.

Therefore, the key issue is the variance of stochastic gradients is not diminishing and fails the LAG rule (4.7) eventually.

4.2 LASG: Lazily Aggregated Stochastic Gradients

In this section, we formally develop our LASG method. To overcome the limitations of LAG in stochastic settings, the key of the LASG design is to **reduce the variance of the innovation measure** appeared in the adaptive condition. As discussed, LASG-WK uses a condition checked by each worker; LASG-PS uses one checked by the parameter server.

Algorithm 4 LASG-WK1

```
Input: Delay counter $\{\tau_m^0\}$, stepsizes $\{\eta_k\}$, max delay $D$.
for $k = 0, 1, \ldots, K-1$ do
    Server broadcasts $\theta^k$ to all workers.
    All workers save $\tilde{\theta} = \theta^k$ if $k \mod D = 0$.
    for $m = 1, 2, \ldots, M$ do in parallel
        Compute $\nabla\ell(\theta^k; \xi_m^k)$, $\nabla\ell(\tilde{\theta}; \xi_m^k)$.
        Check condition (4.9)
        if (4.9) is violated
                or $k \mod D = 0$ then
            Upload $\delta_m^k$.
                ▷ Save $\tilde{\delta}_m^k$ and set $\tau_m^{k+1} = 1$
        else
            Upload nothing.
                ▷ Set $\tau_m^{k+1} = \tau_m^k + 1$
        end if
    end for
    Server updates via (4.4).
end for
```

Algorithm 5 LASG-WK2

```
Input: Delay counter $\{\tau_m^0\}$, stepsizes $\{\eta_k\}$, max delay $D$.
for $k = 0, 1, \ldots, K-1$ do
    Server broadcasts $\theta^k$ to all workers.
    for $m = 1, 2, \ldots, M$ do in parallel
        Compute $\nabla\ell(\theta^k; \xi_m^k)$, $\nabla\ell(\theta_m^{k-\tau_m^k}; \xi_m^k)$.
        Check condition (4.12).
        if (4.12) is violated, or, $\tau_m^k \geq D$ then
            Upload $\delta_m^k$.
                ▷ Save $\theta^k$ and set $\tau_m^{k+1} = 1$
        else
            Upload nothing.
                ▷ Set $\tau_m^{k+1} = \tau_m^k + 1$
        end if
    end for
    Server updates via (4.4).
end for
```

Table 4.2: A comparison of LASG-WK1 and LASG-WK2.

4.2.1 Worker LASG: save communication uploads

We first introduce two LASG-WK variants. The first one, which we term **LASG-WK1**, calculates two stochastic gradient innovations with one at sample ξ_m^k as

$$\tilde{\delta}_m^k := \nabla\ell(\theta^k; \xi_m^k) - \nabla\ell(\tilde{\theta}; \xi_m^k)$$

and one at sample $\xi_m^{k-\tau_m^k}$ as

$$\tilde{\delta}_m^{k-\tau_m^k} := \nabla\ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla\ell(\tilde{\theta}; \xi_m^{k-\tau_m^k}).$$

where $\tilde{\theta}$ is a snapshot of the previous iterate θ that will be updated every D ($\geq d_{\max}$) iterations. As we will show in (4.10), $\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}$ can be viewed as the difference of two variance-reduced gradients calculated at θ^k and $\theta^{k-\tau_m^k}$. Using $\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}$ as the error induced by using stale information, LASG-WK1 excludes m from $\mathcal{M}^k$ if

$$\boxed{\left\|\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\right\|^2 \leq \frac{c}{d_{\max}} \sum_{d=1}^{d_{\max}} \left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2.} \tag{4.9}$$

Recall if (4.9) is satisfied, we increment staleness $\tau_m^{k+1} = \tau_m^k + 1$; otherwise, worker m uploads the fresh stochastic gradient and resets staleness as $\tau_m^{k+1} = 1$.

Behind (4.9) is the reduction of its inherent variance. To see this, decompose the LHS of (4.9) as the difference of two *variance reduced* stochastic gradients at iteration k and $k - \tau_m^k$:

$$\begin{aligned}\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k} = &\left(\nabla\ell(\theta^k; \xi_m^k) - \nabla\ell(\tilde{\theta}; \xi_m^k) + \nabla\mathcal{L}_m(\tilde{\theta})\right) \\ &- \left(\nabla\ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla\ell(\tilde{\theta}; \xi_m^{k-\tau_m^k}) + \nabla\mathcal{L}_m(\tilde{\theta})\right).\end{aligned} \tag{4.10}$$

To provide some intuition, we define the minimizer of (4.2) as $\theta^\star$ and assume that $\nabla\ell(\theta; \xi_m)$ is $\bar{L}$-Lipschitz continuous[1] for any ξ_m. The LHS of (4.9) is *upper-bounded* in expectation by

$$\begin{aligned}\mathbb{E}\left[\left\|\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\right\|^2\right] \leq &\, 8\bar{L}(\mathbb{E}\mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star)) + 8\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k}) - \mathcal{L}(\theta^\star)) \\ &+ 16\bar{L}(\mathbb{E}\mathcal{L}(\tilde{\theta}) - \mathcal{L}(\theta^\star)).\end{aligned} \tag{4.11}$$

[1] This Lipschitz continuous assumption is needed only when we provide some intuitions of our design, but in our subsequent analysis.

When the iterate θ^k converges, e.g., $\theta^k, \theta^{k-\tau_m^k}, \tilde{\theta} \to \theta^*$, the RHS of (4.11) diminishes, and thus the LHS of (4.9) diminishes. This is in contrast to the stochastic LAG-WK rule in (4.8) that is *lower-bounded* by a non-diminishing value.

The second rule **LASG-WK2** excludes m from $\mathcal{M}^k$ if

$$\boxed{\left\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta_m^{k-\tau_m^k}; \xi_m^k)\right\|^2 \leq \frac{c}{d_{\max}} \sum_{d=1}^{d_{\max}} \left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2.} \tag{4.12}$$

Note that different from (4.7), condition (4.12) is evaluated at two different iterates but on the same sample ξ_m^k.

LASG-WK2 (4.12) also reduces its inherent variance since the LHS of (4.12) can be written as the difference between a *variance reduced* stochastic gradient and a *deterministic* gradient, that is

$$\begin{aligned}\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) = \big(&\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) \\ &+ \nabla \mathcal{L}_m(\theta^{k-\tau_m^k})\big) - \nabla \mathcal{L}_m(\theta^{k-\tau_m^k}).\end{aligned} \tag{4.13}$$

With derivations deferred to the supplementary, we conclude that $\mathbb{E}[\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k)\|^2] \to 0$ as $\theta^k \to \theta^\star$.

4.2.2 Server LASG: save up/downloads and calculations

We next introduce two LASG-PS variants. The rationale is that if the model difference is small, the gradient difference used in Chapter 4.2.1 is likely to be small.

The first variant **LASG-PS** excludes m from $\mathcal{M}^k$ if

$$\boxed{L_m^2 \left\|\theta^k - \theta^{k-\tau_m^k}\right\|^2 \leq \frac{c}{d_{\max}} \sum_{d=1}^{d_{\max}} \left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2} \tag{4.14}$$

where L_m is the smoothness constant of $\mathcal{L}_m(\theta)$. Condition (4.14) can be checked at the server side without computing new gradients if the server stores $\{\theta^{k-\tau_m^k}\}$ for each worker m.

The LHS of (4.14) can be *upper-bounded* in expectation by

$$\mathbb{E}\Big[\big\|\theta^k-\theta^{k-\tau_m^k}\big\|^2\Big]\leq 2D\sum_{d=1}^{D}\mathbb{E}\Big[\big\|\theta^{k-d}-\theta^{k-d-\tau_m^{k-d}}\big\|^2\Big]\eta_{k-D}^2$$
$$+\,2D\sum_{d=1}^{D}\mathbb{E}\left\|\nabla\mathcal{L}(\theta^{k-d})\right\|^2\eta_{k-D}^2+D^2\left(\sum_{m\in\mathcal{M}}\sigma_m^2\right)\eta_{k-D}^2. \tag{4.15}$$

Assume $\left\|\nabla\mathcal{L}(\theta^k)\right\|^2$ is bounded; then the diminishing stepsizes $\{\eta_k\}$ ensure that the 2nd and 3rd terms in the RHS of (4.15) vanish. Using mathematical induction, the LHS of (4.14) also diminishes. Therefore, this condition remains effective asymptotically.

When an estimate L_m is not available, one can use LASG-PSE, a variation of LASG-PS that estimates L_m "on-the-fly." With $\hat{L}_m^k$ denoting the estimate of L_m, **LASG-PSE** excludes m from $\mathcal{M}^k$ if

$$\boxed{(\hat{L}_m^k)^2\|\theta^k-\theta^{k-\tau_m^k}\|^2\leq\frac{c}{d_{\max}}\sum_{d=1}^{d_{\max}}\|\theta^{k+1-d}-\theta^{k-d}\|^2} \tag{4.16}$$

where the estimated constant $\hat{L}_m^k$ is updated iteratively via

$$\hat{L}_m^{k+1}=\max\left\{\hat{L}_m^k,\frac{\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)\|}{\|\theta^k-\theta^{k-\tau_m^k}\|}\right\}. \tag{4.17}$$

We give LASG-PS and LASG-PSE in Algorithms 6 and 7, respectively, and compare all LASG methods in Table 4.1.

Comparison of all LASG methods. All the LASG rules can be computed efficiently without storing all previous θ^k. LASG-PS and LASG-PSE need *extra memory* at the server but save both *local computation and download communication* while LASG-WK1 and LASG-WK2 save only upload communication. LASG-WK1 is more conservative as LASG-WK1 measures the change of gradients at two model states for both new and old data samples, but LASG-WK2 measures only the change of gradient at the new sample.

Algorithm 6 LASG-PS

```
Input: $\theta^0$, delay counter $\{\tau_m^0\}$, smoothness contants $\{L_m\}$, stepsizes $\{\eta_k\}$, maximum delay $D$.
for $k = 0, 1, \ldots, K-1$ do
    for $m = 1, 2, \ldots, M$ do in parallel
        Server checks condition (4.14).
        if (4.14) is violated or $\tau_m^k \geq D$ then
            Server sends $\theta^k$ to worker $m$
            Worker $m$ computes $\nabla \ell(\theta^k; \xi_m^k)$.
            Worker $m$ uploads $\delta_m^k$.
                        ▷ Save $\theta^k$ and $\tau_m^{k+1} = 1$
        else
            No action.    ▷ $\tau_m^{k+1} = \tau_m^k + 1$
        end if
    end for
    Server updates via (4.4).
end for
```

Algorithm 7 LASG-PSE

```
Input: $\theta^0$, delay counter $\{\tau_m^0\}$, smoothness estimates $\{\hat{L}_m^0\}$, stepsizes $\{\eta_k\}$, maximum delay $D$.
for $k = 0, 1, \ldots, K-1$ do
    for $m = 1, 2, \ldots, M$ do in parallel
        Server checks condition (4.16).
        if (4.16) is violated or $\tau_m^k \geq D$ then
            Server sends $\theta^k$ to worker $m$.
            Worker $m$ computes $\nabla \ell(\theta^k; \xi_m^k)$.
            Worker $m$ uploads $\delta_m^k$.
                        ▷ Save $\theta^k$ and $\tau_m^{k+1} = 1$
            Worker $m$ uploads $\hat{L}_m^{k+1}$ in (4.17).
        else
            No action.    ▷ $\tau_m^{k+1} = \tau_m^k + 1$
        end if
    end for
    Server updates via (4.4).
end for
```

Table 4.3: A comparison of LASG-PS and LASG-PSE.

4.2.3 Quantized LASG: Save also communication bits

We further reduce communication bits per round by applying quantization.We define the gradient under a quantization operator $\mathcal{Q}$ as

$$Q(\theta; \xi) := \mathcal{Q}\left(\nabla \ell(\theta; \xi)\right). \tag{4.18}$$

We adopt the stochastic quantization scheme in [2] and develop *quantized LASG* (LAQSG) as

$$\theta^{k+1} = \theta^k - \eta_k \sum_{m \in \mathcal{M} \backslash \mathcal{M}^k} Q(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \eta_k \sum_{m \in \mathcal{M}^k} Q(\theta^k; \xi_m^k)$$

where $\mathcal{M}^k$ is determined by one of four described rules.

4.3 Main Results

In this section we present the convergence results of LASG-WK1, LASG-WK2 and LASG-PS under both the nonconvex condition and the Polyak-Łojasiewicz condition, and the convergence results of LAQSG under the nonconvex condition only. We leave the analysis of LASG-PSE for future work, but it empirically has very impressive performance.

First, we make some basic assumptions.

Assumption 1 *The loss function $\mathcal{L}(\theta)$ is L-smooth, i.e. for any $\theta_1, \theta_2 \in \mathbb{R}^p$, it follows that*

$$\mathcal{L}(\theta_2) \leq \mathcal{L}(\theta_1) + \langle \nabla \mathcal{L}(\theta_1), \theta_2 - \theta_1 \rangle + \frac{L}{2} \|\theta_2 - \theta_1\|^2. \tag{4.19}$$

Assumption 2 *The samples $\xi_m^1, \xi_m^2, \ldots$ are independent, and the stochastic gradient $\nabla \ell(\theta; \xi_m^k)$ satisfies*

$$\mathbb{E}_{\xi_m^k} \left[\nabla \ell(\theta; \xi_m^k) \right] = \nabla \mathcal{L}_m(\theta), \tag{4.20a}$$

$$\mathbb{E}_{\xi_m^k} \left[\|\nabla \ell(\theta; \xi_m^k) - \nabla \mathcal{L}_m(\theta)\|^2 \right] \leq \sigma_m^2. \tag{4.20b}$$

For LASG-PS, we require an extra smoothness assumption.

Assumption 3 *The local gradient $\nabla \mathcal{L}_m$ is L_m-Lipschitz continuous, i.e. for any $\theta_1, \theta_2 \in \mathbb{R}^p$, we have*

$$\|\nabla \mathcal{L}_m(\theta_1) - \nabla \mathcal{L}_m(\theta_2)\| \leq L_m \|\theta_1 - \theta_2\|. \tag{4.21}$$

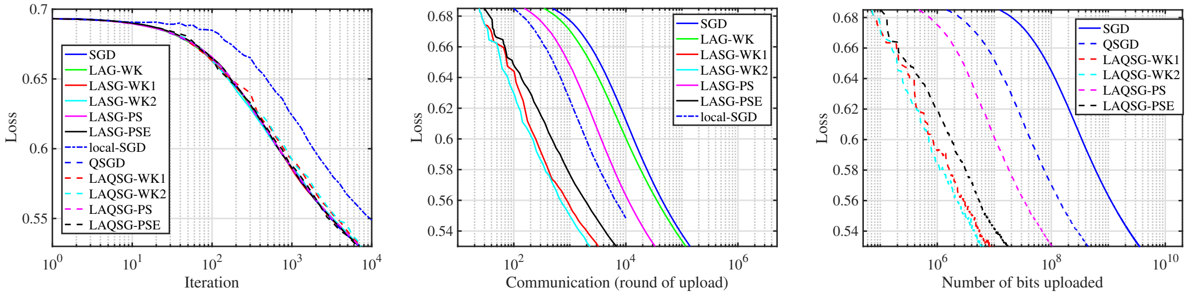

Figure 4.3: Logistic regression on *covtype* dataset

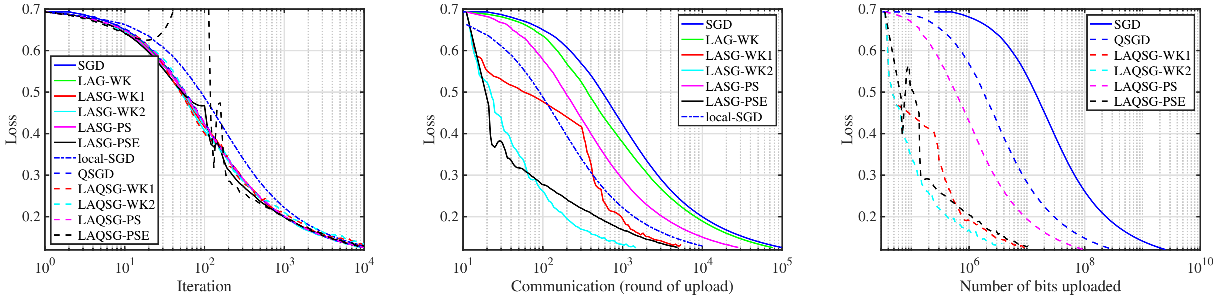

Figure 4.4: Logistic regression on *mnist* digits 3 and 5

Assumption 1 implies that the loss function $\mathcal{L}$ can be upper bounded by a quadratic function at any point. Assumption 2 ensures that the stochastic gradient is unbiased, and has bounded variance. Assumption 3 bounds the change of local gradients when they are evaluated at two points. Assumptions 1-3 are common in analyzing SGD [30, 47, 2, 105, 3, 119, 114, 109].

4.3.1 Convergence in the nonconvex case

We first present the convergence in the nonconvex case.

Theorem 4 (nonconvex) *Under Assumptions 1, 2 (for Algorithm 6 also Assumption 3), if the stepsize is chosen as* $\eta_k = \eta = \mathcal{O}(\frac{1}{\sqrt{K}})$*, and the threshold is* $c \leq \min\{\frac{1}{12D\eta^2}, \frac{\sqrt{M}L^2}{18}\}$*, then*

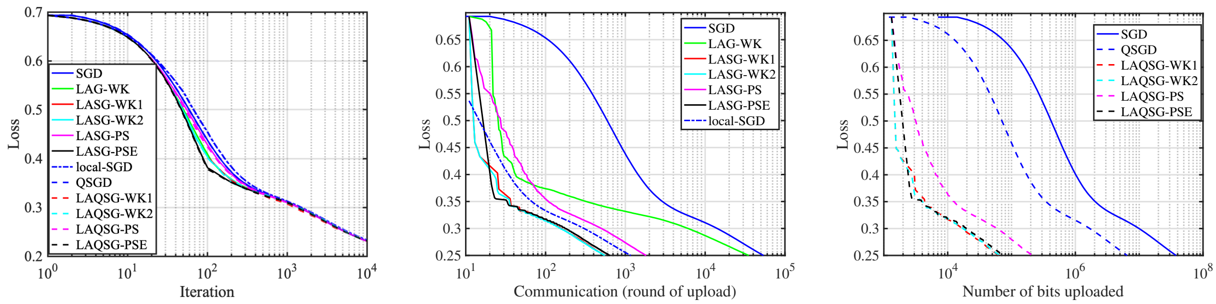

Figure 4.5: Logistic regression on *ijcnn1* dataset

$\{\theta^k\}$ *generated by Algorithms 4-6 satisfy*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] = \mathcal{O}\left(\frac{\sqrt{M}}{K} + \frac{\sqrt{\sum_{m=1}^{M}\sigma_m^2}}{M^{\frac{3}{4}}\sqrt{K}}\right). \tag{4.22}$$

From Theorem 4, the convergence rate of LASG in terms of the average gradient norms is still $\mathcal{O}(1/\sqrt{K})$, matching standard SGD [30]. When $K \gg M$, the second term is dominant. If we simplify $\sigma_m = \sigma$, $\forall m$, then the bound becomes $\mathcal{O}(1/(M^{\frac{1}{4}}K^{\frac{1}{2}}))$, and the convergence rate will be improved as the number of workers M increases.

The assumption below bounds the variance of the quantized stochastic gradient.

Assumption 4 *For any $\theta \in \mathbb{R}^p$ and any $m \in \mathcal{M}$, we have $\mathbb{E}_{\xi_m}\left[\|\nabla\ell(\theta;\xi_m)\|^2\right] \le B$.*

Based on this assumption, we have the following result.

Theorem 5 (LAQSG) *Under Assumptions 1, 2, 4 (also Assumption 3 for Algorithm 6), if $\eta_k = \eta = \mathcal{O}(\frac{1}{\sqrt{K}})$, $c \le \min\{\frac{d_{\max}}{16D\eta^2}, \frac{d_{\max}\sqrt{M}L^2}{24}\}$ where $c_\eta > 0$ is a constant, then $\{\theta^k\}$ generated by quantized Algorithms 4 - 6 satisfy*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] = \mathcal{O}\left(1/\sqrt{K}\right). \tag{4.23}$$

The rate $\mathcal{O}(1/\sqrt{K})$ matches the standard QSGD [2].

4.3.2 Convergence under the Polyak-Łojasiewicz condition

Assumption 5 *The loss function $\mathcal{L}$ satisfies the Polyak-Łojasiewicz (PL) condition with constant $\mu > 0$, that is*

$$\mathcal{L}(\theta) - \mathcal{L}(\theta^{*}) \leq \frac{1}{2\mu} \left\| \mathcal{L}(\theta) \right\|^{2}. \tag{4.24}$$

The PL condition is weaker than strong convexity and may hold with convexity [51]. It is met by underdetermined least squares and logistic regression.

Theorem 6 (PL-condition) *Under Assumption 1,2,5 (for Algorithm 6 also Assumption 3), if $\eta_k = \frac{2}{\mu(k+K_0)} \leq \eta_0$ for a given constant K_0, and $c \leq \min\{\frac{d_{\max}}{24D\eta_0^2}, \frac{d_{\max}\sqrt{M}L^2}{18}\}$, then θ^K generated by Algorithms 4, 5 and 6 satisfies*

$$\mathbb{E}\big[\mathcal{L}(\theta^{K})\big] - \mathcal{L}(\theta^{\star}) = \mathcal{O}\left(1/K\right). \tag{4.25}$$

The rate $\mathcal{O}(1/K)$ matches that of SGD [89]. Under the same (or even slightly stronger) assumptions of Theorem 3, it has been shown that $\mathcal{O}(1/K)$ is the best rate by any stochastic gradient-based algorithm; see [78, Theorems 5.3.1 and 7.2.6].

4.4 Numerical Tests

We conducted numerical tests on both logistic regression and neural network models. We benchmarked LA(Q)SG with SGD, LAG-WK, local SGD (with varying intervals H) and QSGD. We did a grid search for best learning rates.

4.4.1 Logistic regression

The data are distributed across $M = 10$ workers for ijcnn1, MNIST (with digits 3, 5) and $M = 20$ for Covtype. For each worker, the batch size is selected to be 0.01 of the local data size for ijcnn1, MNIST and 0.001 for Covtype. The ℓ_2-regularization parameter is set to

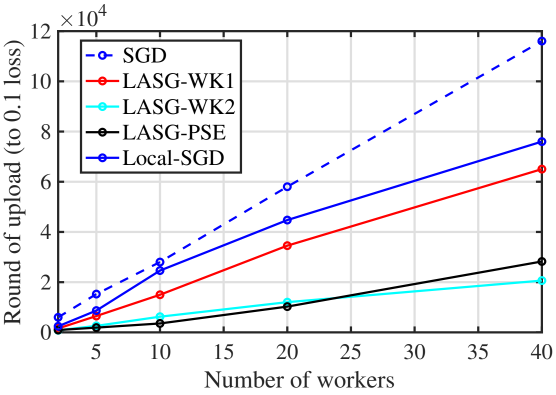

Figure 4.6: Training loss on *mnist* dataset under different number of workers.

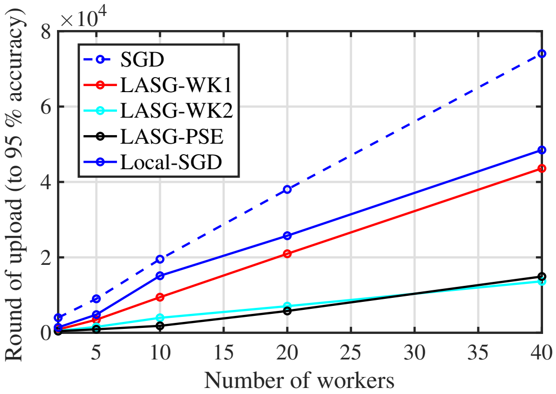

Figure 4.7: Test accuracy on *mnist* dataset under different number of workers.

be 10^{-5}. We choose stepsize $\eta = 0.1$. For all LASG algorithms, $D = 100$, $d_{\max} = 10$ and $c = 1/\eta^2$. For local-SGD, the communication period is $H = 50, 10, 20$ iterations for ijcnn1, MNIST, Covtype respectively. This is optimized to save communication as much as possible without largely affecting the convergence speed. For quantization methods, we perform 4-bit stochastic quantization [2]. Numerical results are reported in Figures 4.3-4.5.

4.4.2 Training neural networks

We train a convolutional neural network with two convolution-ELU-maxpooling layers (ELU is a smoothed ReLU) followed by two fully-connected layers for 10 classes classification on MNIST. The data are distributed on $M = 10$ workers. We choose stepsize $\eta = 0.05$. Since

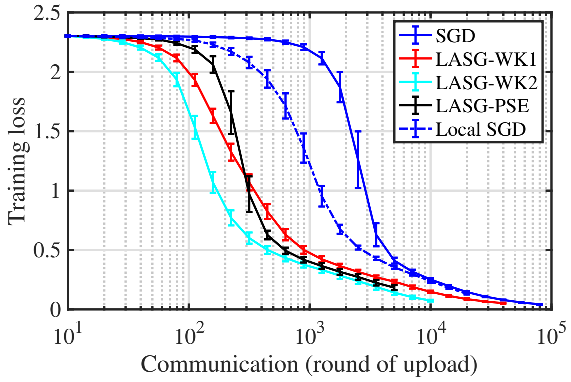

Figure 4.8: Training loss on *mnist* dataset averaged over 30 trials.

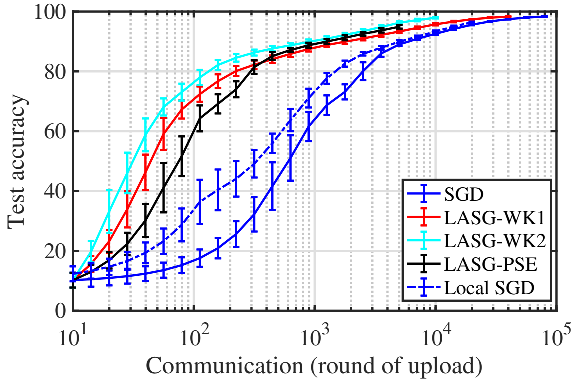

Figure 4.9: Test accuracy on *mnist* dataset averaged over 30 trials.

the objective function is nonsmooth (L_m is unavailable), LASG-PS is not tested. For other LASG algorithms, we set $D = 50$, $d_{\max} = 10$, and $c = 1/\eta^2$. For local-SGD, we set $H = 4$. For all quantization methods, we set 8 bits. We first report the the total number of uploads needed to achieve the training loss 0.1 and the test accuracy 95% under different number of workers M in Figures 4.6 and 4.7, respectively. We also report the numerical results averaged over 30 Monte Carlo runs in Figures 4.8 and 4.9.

All algorithms have been tested on the popular *tiny imagenet* dataset, which contains 200 classes and 500 images per class for training and 10,000 images for testing. All images in *tiny imagenet* are 64x64 colored ones. We use the Resnet18 model initialized by weights pretrained on ImageNet1000; see the accuracy versus the number of communication uploads in Figures 4.10 and 4.11. For training loss, LASG-WK1 and -WK2 require much less total

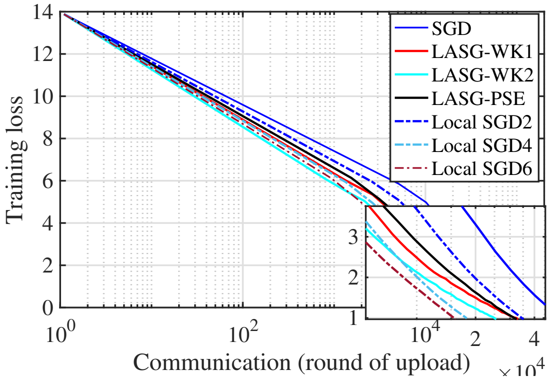

Figure 4.10: Training loss on *tiny imagenet* dataset.

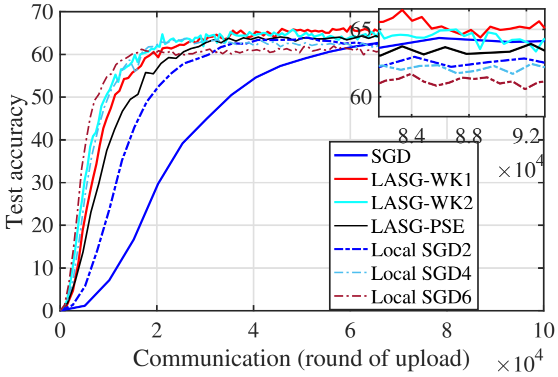

Figure 4.11: Test accuracy on *tiny imagenet* dataset.

time than SGD and local SGD with $H = 2$, but slightly more than local SGD with $H = 4$ and 6. However, as shown in Figure 4.11, local SGD with larger communication period sacrifices the testing accuracy by 3-4%. This reduced test accuracy is common among local SGD methods, which has been studied theoretically; see e.g., [126].

All LASG algorithms has the same iteration complexity as SGD and outperform local-SGD in most cases. Compared with SGD, LASG-WK2 and LASG-PSE reduce communication rounds by about one order of magnitude for neural network training and even more for logistic regression. LASG-WK1 also reduce communication by more than one order of magnitude for logistic regression. Based on the results of LAG-WK, it is evident that the selection rules (4.9), (4.12) and (4.16) achieve more significant improvement in terms of saving communication than the selection rule (4.7) of LAG-WK. Moreover, the performance of LAQSG validates that LASG can be easily equipped with stochastic quantization with extra benefits from quantization.

4.5 Appendix

We first highlight the key steps, present some supporting lemmas that will be used frequently in the subsequent analysis, which is followed by the proofs of the results in Chapter 4.3.

4.5.1 Derivations of missing steps in Chapter 4.2

We will provide the detailed derivations of some missing steps in Chapter 4.2. We define an auxiliary function as

$$\psi_m(\theta) := \mathcal{L}_m(\theta) - \mathcal{L}_m(\theta^\star) - \langle \nabla \mathcal{L}_m(\theta^\star), \theta - \theta^\star \rangle$$

where $\theta^\star$ is a minimizer of $\mathcal{L}(\theta)$. Assume that $\nabla \ell(\theta; \xi_m)$ is $\bar{L}$-Lipschitz continuous for all ξ_m, we have $\|\nabla \ell(\theta; \xi_m) - \nabla \ell(\theta^\star; \xi_m)\|^2 \leq 2\bar{L}(\ell(\theta; \xi_m) - \ell(\theta^\star; \xi_m) - \langle \nabla \ell(\theta^\star; \xi_m), \theta - \theta^\star \rangle)$. Taking expectation with respect to ξ_m, we can obtain

$$\begin{aligned}&\mathbb{E}_{\xi_m}[\|\nabla \ell(\theta; \xi_m) - \nabla \ell(\theta^\star; \xi_m)\|^2] \leq \\ &2\bar{L}\left(\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta^\star) - \langle \nabla \mathcal{L}_m(\theta^\star), \theta - \theta^\star \rangle\right) = 2\bar{L}\psi_m(\theta).\end{aligned} \tag{4.26}$$

Note that $\nabla \mathcal{L}_m$ is also $\bar{L}$-Lipschitz continuous and thus

$$\|\nabla \mathcal{L}_m(\theta) - \nabla \mathcal{L}_m(\theta^\star)\|^2 \leq 2\bar{L}\psi_m(\theta).$$

Derivations of (4.8)

By (4.38), we can derive that $\|\theta_1\|^2 \geq \frac{1}{2}\|\theta_1 + \theta_2\|^2 - \|\theta_2\|^2$. As a consequence, we can obtain

$$\begin{aligned}
&\mathbb{E}\Big[\big\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\big\|^2\Big] \\
\geq& \frac{1}{2}\mathbb{E}\Big[\big\|\big(\nabla \ell(\theta^k; \xi_m^k) - \nabla \mathcal{L}_m(\theta^k)\big) \\
&+ \big(\nabla \mathcal{L}_m(\theta^{k-\tau_m^k}) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\big)\big\|^2\Big] \\
&- \mathbb{E}\Big[\big\|\nabla \mathcal{L}_m(\theta^k) - \nabla \mathcal{L}_m(\theta^{k-\tau_m^k})\big\|^2\Big] \\
=& \frac{1}{2}\mathbb{E}\Big[\big\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \mathcal{L}_m(\theta^k)\big\|^2\Big] \\
&+ \frac{1}{2}\mathbb{E}\Big[[\big\|\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \mathcal{L}_m(\theta^{k-\tau_m^k})\big\|^2]\Big] \\
&+ \mathbb{E}\Big[\langle \nabla \ell(\theta^k; \xi_m^k) - \nabla \mathcal{L}_m(\theta^k), \nabla \mathcal{L}_m(\theta^{k-\tau_m^k}) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\rangle\Big] \\
&- \mathbb{E}\Big[\big\|\nabla \mathcal{L}_m(\theta^k) - \nabla \mathcal{L}_m(\theta^{k-\tau_m^k})\big\|^2\Big].
\end{aligned}$$

To obtain (4.8), we use that

$$\Big\langle \underbrace{\mathbb{E}\big[\nabla \ell(\theta^k; \xi_m^k)\big|\Theta^k\big]}_{=\nabla \mathcal{L}_m(\theta^k)} - \nabla \mathcal{L}_m(\theta^k), \nabla \mathcal{L}_m(\theta^{k-\tau_m^k}) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\Big\rangle = 0.$$

Derivations of (4.11)

Recall that

$$\begin{aligned}
&\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k} \\
=& \big(\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\tilde{\theta}; \xi_m^k) + \nabla \mathcal{L}_m(\tilde{\theta})\big) \\
&- \big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\tilde{\theta}; \xi_m^{k-\tau_m^k}) + \nabla \mathcal{L}_m(\tilde{\theta})\big) \\
=& \underbrace{\big(\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\tilde{\theta}; \xi_m^k) + \nabla \psi_m(\tilde{\theta})\big)}_{:=g_m^k} \\
&- \underbrace{\big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\tilde{\theta}; \xi_m^{k-\tau_m^k}) + \nabla \psi_m(\tilde{\theta})\big)}_{:=g_m^{k-\tau_m^k}}.
\end{aligned}$$

And by (4.38), we have $\|\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\|^2 \leq 2\|g_m^k\|^2 + 2\|g_m^{k-\tau_m^k}\|^2$. We decompose the first term as

$$\begin{aligned}
\mathbb{E}[\|g_m^k\|^2] \leq & 2\mathbb{E}[\|\nabla\ell(\theta^k;\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k)\|^2] \\
& + 2\mathbb{E}[\|\nabla\ell(\tilde{\theta};\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k) - \nabla\psi_m(\tilde{\theta})\|^2] \\
= & 2\mathbb{E}[\mathbb{E}[\|\nabla\ell(\theta^k;\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k)\|^2|\Theta^k]] \\
& + 2\mathbb{E}[\|\nabla\ell(\tilde{\theta};\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k) \\
& - \mathbb{E}[\nabla\ell(\tilde{\theta};\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k)|\Theta^k]\|^2] \\
\leq & 4\bar{L}\mathbb{E}[\psi_m(\theta^k)] + 2\mathbb{E}[\|\nabla\ell(\tilde{\theta};\xi_m^k) - \nabla\ell(\theta^\star;\xi_m^k)\|^2] \\
\overset{(a)}{\leq} & 4\bar{L}\mathbb{E}[\psi_m(\theta^k)] + 4\bar{L}\mathbb{E}\psi_m(\tilde{\theta}).
\end{aligned}$$

where (a) follows from (4.26).

By nonnegativity of ψ_m, we have

$$\begin{aligned}
\mathbb{E}[\|g_m^k\|^2] &\leq 4\bar{L}\sum_{m\in\mathcal{M}}\mathbb{E}\psi_m(\theta^k) + 4\bar{L}\sum_{m\in\mathcal{M}}\mathbb{E}\psi_m(\tilde{\theta}) \\
&= 4M\bar{L}(\mathbb{E}\mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star)) + 4M\bar{L}(\mathbb{E}\mathcal{L}(\tilde{\theta}) - \mathcal{L}(\theta^\star)). \qquad (4.27)
\end{aligned}$$

Similarly, we can prove

$$\mathbb{E}[\|g_m^{k-\tau_m^k}\|^2] \leq 4M\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k}) - \mathcal{L}(\theta^\star)) + 4M\bar{L}(\mathbb{E}\mathcal{L}(\tilde{\theta}) - \mathcal{L}(\theta^\star)). \qquad (4.28)$$

Therefore, it follows that

$$\begin{aligned}
\mathbb{E}[\|\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\|^2] \leq & 8M\bar{L}(\mathbb{E}\mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star)) \\
& + 8M\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k}) - \mathcal{L}(\theta^\star)) + 16M\bar{L}(\mathbb{E}\mathcal{L}(\tilde{\theta}) - \mathcal{L}(\theta^\star)).
\end{aligned}$$

Derivations of (4.13)

The LHS of (4.12) can be written as

$$\begin{aligned}
&\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)\\
=&\big(\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)+\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})\big)-\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})\\
=&\big(\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)+\nabla\psi_m(\theta^{k-\tau_m^k})\big)-\nabla\psi_m(\theta^{k-\tau_m^k}).
\end{aligned}$$

Similar to (4.27), we can obtain

$$\begin{aligned}
&\mathbb{E}[\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)+\nabla\psi_m(\theta^{k-\tau_m^k})\|^2]\\
&\leq 4M\bar{L}(\mathbb{E}\mathcal{L}(\theta^k)-\mathcal{L}(\theta^\star))+4M\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k})-\mathcal{L}(\theta^\star)).
\end{aligned}$$

Combined with the fact

$$\begin{aligned}
\mathbb{E}[\|\nabla\psi_m(\theta^{k-\tau_m^k})\|^2]&=\mathbb{E}[\|\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})-\nabla\mathcal{L}_m(\theta^\star)\|^2]\\
&\leq 2\bar{L}\mathbb{E}\psi_m(\theta^{k-\tau_m^k})\\
&\leq 2M\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k})-\mathcal{L}(\theta^\star))
\end{aligned}$$

we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^k)\|^2]\\
\leq&8M\bar{L}(\mathbb{E}\mathcal{L}(\theta^k)-\mathcal{L}(\theta^\star))+12M\bar{L}(\mathbb{E}\mathcal{L}(\theta^{k-\tau_m^k})-\mathcal{L}(\theta^\star)).
\end{aligned}$$

Derivations of (4.15)

Expanding LASG update, we have

$$\begin{aligned}
&\mathbb{E}\Big[\|\theta^k-\theta^{k-\tau_m^k}\|^2\Big]\\
=&\frac{1}{M^2}\,\mathbb{E}\Big[\Big\|\sum_{d=1}^{\tau_m^k}\sum_{m\in\mathcal{M}}\eta_{k-d}\nabla\ell(\theta^{k-d-\tau_m^{k-d}};\xi_m^{k-d-\tau_m^{k-d}})\Big\|^2\Big]\\
\leq&\frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\eta_{k-d}^2\,\mathbb{E}\Big[\Big\|\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-d-\tau_m^{k-d}};\xi_m^{k-d-\tau_m^{k-d}})\Big\|^2\Big]
\end{aligned}$$

where we used the Cauchy-Schwartz inequality.

Using $\mathbb{E}[\|A-\mathbb{E}[A]\|^2]+\|\mathbb{E}[A]\|^2=\mathbb{E}[\|A\|^2]$, we have

$$\begin{aligned}
\mathbb{E}\Big[\|\theta^k-\theta^{k-\tau_m^k}\|^2\Big] &= \frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\eta_{k-d}^2 \\
&\times \mathbb{E}\Big[\Big\|\sum_{m\in\mathcal{M}}\Big(\nabla\ell(\theta^{k-d-\tau_m^{k-d}};\xi_m^{k-d-\tau_m^{k-d}})-\nabla\mathcal{L}_m(\theta^{k-d-\tau_m^{k-d}})\Big)\Big\|^2\Big] \\
&+\frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\eta_{k-d}^2\mathbb{E}\Big[\Big\|\sum_{m\in\mathcal{M}}\nabla\mathcal{L}_m(\theta^{k-d-\tau_m^{k-d}})\Big\|^2\Big] \\
&\leq\frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\eta_{k-d}^2\sum_{m\in\mathcal{M}}\sigma_m^2+\frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\eta_{k-d}^2\mathbb{E}\Big[\Big\|\sum_{m\in\mathcal{M}}\nabla\mathcal{L}_m(\theta^{k-d-\tau_m^{k-d}})\Big\|^2\Big] \\
&\leq\frac{\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\sum_{m\in\mathcal{M}}\sigma_m^2\eta_{k-d}^2+\frac{2\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\mathbb{E}\Big[\Big\|\nabla\mathcal{L}(\theta^{k-d})\Big\|^2\Big]\eta_{k-d}^2 \\
&+\frac{2\tau_m^k}{M^2}\sum_{d=1}^{\tau_m^k}\sum_{m\in\mathcal{M}}L_m^2\mathbb{E}\Big[\Big\|\theta^{k-d}-\theta^{k-d-\tau_m^{k-d}}\Big\|^2\Big]\eta_{k-d}^2.
\end{aligned}$$

We arrive at our statement since $\tau_m^k\leq D$ and $\eta_{k-d}\leq\eta_{k-D}$.

4.5.2 Key steps of Lyapunov analysis

With these assumptions, LASG will yield descent of $\mathcal{L}(\theta^k)$.

Lemma 11 *Under Assumptions 1, 2 and 3, $\{\theta^k\}$ generated by Algorithms 4, 5 and 6 satisfy*

$$\begin{aligned}
\mathbb{E}[\mathcal{L}(\theta^{k+1})]-\mathbb{E}[\mathcal{L}(\theta^k)] \leq & -\left(\eta_k-\frac{L\eta_k^2}{2}\right)\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] \\
&+\ \frac{L\eta_k^2}{2}\mathbb{E}\left[\left\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})-\nabla\mathcal{L}(\theta^k)\right\|^2\right] \\
&+\frac{(\eta_k-L\eta_k^2)}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\Big[\langle\nabla\mathcal{L}(\theta^k),\delta_m^k\rangle\Big] \qquad (4.29)
\end{aligned}$$

where $\delta_m^k:=\nabla\ell(\theta^k;\xi_m^k)-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})$.

Among three terms in the RHS of (4.29): the first term resembles the standard unbiased stochastic descent; the second term captures the variance of the *stale* aggregated stochastic gradient; and, the last term quantifies the correlations between the gradient direction $\nabla\mathcal{L}(\theta^k)$ and the error induced by the *stale* stochastic gradient ∇^k.

Analyzing the progress of $\mathcal{L}(\theta^k)$ under LASG is challenging. Below we characterize the regularity of the stale stochastic gradients ∇^k, which lays the foundation for incorporating the properly controlled staleness into the SGD update.

Lemma 12 *Under Assumptions 1 and 2, if the stepsizes satisfy $\eta_{k+1} \le \eta_k \le 1/L$, then we have*

$$\mathbb{E}\Big[\langle\nabla\mathcal{L}(\theta^k), \delta_m^k\rangle\Big] \le \frac{L\eta_k}{2}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2\right] + \frac{6DL\eta_k}{2M\sqrt{M}}\sum_{m\in\mathcal{M}}\sigma_m^2$$
$$+\sum_{d=1}^{D}\left(\frac{c}{2L\eta_k d_{\max}} + \frac{\sqrt{M}L}{12\eta_k}\right)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right]. \tag{4.30}$$

Lemma 12 justifies the relevance of the stale yet properly selected stochastic gradients. Intuitively, the first term in the RHS of (4.30) will reduces the magnitude of descent in (4.29), and the second and third terms will diminish if the stepsizes are diminishing since $\mathbb{E}\left[\|\theta^k - \theta^{k-1}\|^2\right] = \mathcal{O}(\eta_k^2)$.

The next lemma implies that the variance of the *stale* aggregated stochastic gradient reduces to that of standard SGD if the stepsizes are diminishing since $\mathbb{E}\left[\|\theta^k - \theta^{k-1}\|^2\right] = \mathcal{O}(\eta_k^2)$.

Lemma 13 *Under Assumptions 1 and 2, if the stepsizes satisfy $\eta_{k+1} \le \eta_k \le 1/L$, then we have*

$$\mathbb{E}\left[\left\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}) - \nabla\mathcal{L}(\theta^k)\right\|^2\right]$$
$$\le\frac{3c}{d_{\max}}\sum_{d=1}^{d_{\max}}\mathbb{E}\|\theta^{k+1-d} - \theta^{k-d}\|^2 + \frac{9}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2. \tag{4.31}$$

In view of Lemmas 11-13, we introduce the following **Lyapunov function** to capture the progress of LASG:

$$V^k := \mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star) + \sum_{d=1}^{D} \gamma_d \|\theta^{k+1-d} - \theta^{k-d}\|^2 \tag{4.32}$$

where $\{\gamma_d\}_{d=1}^D$ are constants to be determined later. The following lemma is a direct application of Lemmas 11–13.

Lemma 14 *Under Assumptions 1 and 2, there exist nonnegative constants $\{A_d^k\}_{d=1}^D$, B_0^k and B_1^k such that*

$$\begin{aligned}\mathbb{E}[V^{k+1}] - \mathbb{E}[V^k] \leq &- B_0^k \,\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] + B_1^k \sum_{m=1}^{M} \sigma_m^2 \\ &- \sum_{d=1}^{D} A_d^k \,\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right].\end{aligned} \tag{4.33}$$

The constants $\{A_d^k\}_{d=1}^D$, B_0^k and B_1^k depend on the stepsize η_k, the threshold c and the parameters $\{\gamma_d\}_{d=1}^D$. Their expressions are specified in the proof. By choosing proper η_k and c, we are able to ensure the convergence of LASG.

4.5.3 Supporting lemmas

Define the σ-algebra $\Theta^k = \{\theta^l, 1 \leq l \leq k\}$. For convenience, we initialize parameters as $\theta^{-D} = \cdots = \theta^{-1} = \theta^0$, and define the difference between θ^{k+1-d} and θ^{k-d} as

$$\Delta^{k-d} := \theta^{k+1-d} - \theta^{k-d} \tag{4.34}$$

which implies that $\Delta^k := \theta^{k+1} - \theta^k$.

Some basic facts used in the proof are reviewed as follows.

Fact 1. Assume that $X_1, X_2, \ldots, X_n \in \mathbb{R}^p$ are independent random variables, and $EX_1 = \cdots = EX_n = 0$. Then

$$\mathbb{E}\left[\left\|\sum_{i=1}^{n} X_i\right\|^2\right] = \sum_{i=1}^{n} \mathbb{E}\left[\|X_i\|^2\right]. \tag{4.35}$$

Fact 2. (Young's inequality) For any $\theta_1, \theta_2 \in \mathbb{R}^p, \varepsilon > 0$,

$$\langle \theta_1, \theta_2 \rangle \leq \frac{\|\theta_1\|^2}{2\varepsilon} + \frac{\varepsilon\|\theta_2\|^2}{2}. \tag{4.36}$$

As a consequence, we have

$$\|\theta_1 + \theta_2\|^2 \leq \Big(1 + \frac{1}{\varepsilon}\Big)\|\theta_1\|^2 + (1+\varepsilon)\|\theta_2\|^2. \tag{4.37}$$

Fact 3. (Cauchy-Schwartz) For $\theta_1, \ldots, \theta_n \in \mathbb{R}^p$, we have

$$\Big\|\sum_{i=1}^{n} \theta_i\Big\|^2 \leq n \sum_{i=1}^{n} \|\theta_i\|^2. \tag{4.38}$$

Lemma 15 *For $k - D \leq l \leq k - \tau_m^k$, if $\{\theta^k\}$ are the iterates generated by LASG, we have*

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^k), \nabla\ell(\theta^l; \xi_m^k) - \nabla\ell(\theta^l; \xi_m^{k-\tau_m^k})\rangle\right] \\
\leq& \frac{\sqrt{M}L}{12\eta_k}\sum_{d=1}^{D}\mathbb{E}\left[\|\Delta^{k-d}\|^2\right] + \frac{6DL\eta_k}{\sqrt{M}}\sigma_m^2
\end{aligned} \tag{4.39}$$

and similarly, we have

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^k), \nabla\mathcal{L}_m(\theta^l) - \nabla\ell(\theta^l; \theta^{k-\tau_m^k})\rangle\right] \\
\leq& \frac{\sqrt{M}L}{12\eta_k}\sum_{d=1}^{D}\mathbb{E}\left[\|\Delta^{k-d}\|^2\right] + \frac{3DL\eta_k}{\sqrt{M}}\sigma_m^2.
\end{aligned} \tag{4.40}$$

Proof: We first prove (4.39) by decomposing it as

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^k), \nabla\ell(\theta^l; \xi_m^k) - \nabla\ell(\theta^l; \xi_m^{k-\tau_m^k})\rangle\right] \\
\overset{\text{(a)}}{=}&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^k) - \nabla\mathcal{L}(\theta^l), \nabla\ell(\theta^l; \xi_m^k) - \nabla\ell(\theta^l; \xi_m^{k-\tau_m^k})\rangle\right] \\
\leq& L\mathbb{E}\left[\|\theta^k - \theta^l\|\|\nabla\ell(\theta^l; \xi_m^k) - \nabla\ell(\theta^l; \xi_m^{k-\tau_m^k})\|\right] \\
\overset{\text{(b)}}{\leq}& \frac{\sqrt{M}L}{12D\eta_k}\underbrace{\mathbb{E}\left[\|\theta^k - \theta^l\|^2\right]}_{:=T_1} + \frac{6DL\eta_k}{2\sqrt{M}}\underbrace{\mathbb{E}\left[\|\nabla\ell(\theta^l; \xi_m^k) - \nabla\ell(\theta^l; \xi_m^{k-\tau_m^k})\|^2\right]}_{:=T_2}
\end{aligned} \tag{4.41}$$

where (a) holds due to

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^l), \nabla\ell(\theta^l;\xi_m^k) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\rangle\right]\\
=&\mathbb{E}\left[\mathbb{E}\Big[\langle \nabla\mathcal{L}(\theta^l), \nabla\ell(\theta^l;\xi_m^k) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\rangle\Big|\Theta^l\Big]\right]\\
=&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^l), \mathbb{E}\Big[\nabla\ell(\theta^l;\xi_m^k) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big|\Theta^l\Big]\rangle\right]\\
=&\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^l), \nabla\mathcal{L}_m(\theta^l) - \nabla\mathcal{L}_m(\theta^l)\rangle\right] = 0
\end{aligned}$$

and (b) is a direct application of Fact 2.

Applying Fact 3 to T_1, we have

$$T_1 = \mathbb{E}\Big[\Big\|\sum_{d=1}^{k-l}\Delta^{k-d}\Big\|^2\Big] \leq (k-l)\sum_{d=1}^{k-l}\mathbb{E}\Big[\|\Delta^{k-d}\|^2\Big] \leq D\sum_{d=1}^{D}\mathbb{E}\Big[\|\Delta^{k-d}\|^2\Big] \tag{4.42}$$

and applying Fact 1 to T_2, we have

$$\begin{aligned}
T_2 =&\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^k) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big\|^2\Big]\\
=&\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^k) - \nabla\mathcal{L}_m(\theta^l) + \nabla\mathcal{L}_m(\theta^l) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big\|^2\Big]\\
=&\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^k) - \nabla\mathcal{L}_m(\theta^l)\big\|^2\Big] + \mathbb{E}\Big[\big\|\nabla\mathcal{L}_m(\theta^l) - \nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big\|^2\Big]\\
\leq&2\sigma_m^2
\end{aligned} \tag{4.43}$$

where the last inequality uses Assumption 2. Plugging (4.42) and (4.43) into (4.41), it leads to (4.39).

Likewise, following the steps to (4.41), it can be verified that (4.40) also holds true.

4.5.4 Proof of Lemma 11

Due to the smoothness of $\mathcal{L}(\theta)$ in Assumption 1, we have

$$\begin{aligned}
&\mathbb{E}\big[\mathcal{L}(\theta^{k+1})\big] - \mathbb{E}\big[\mathcal{L}(\theta^{k})\big] \\
\leq &\eta_k \underbrace{\mathbb{E}\Big[-\langle \nabla \mathcal{L}(\theta^k), \frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\rangle\Big]}_{:=I_1} \\
&+ \frac{L\eta_k^2}{2} \underbrace{\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\Big\|^2\Big]}_{:=I_2}.
\end{aligned} \tag{4.44}$$

With $\delta_m^k := \nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})$ denoting the stochastic gradient innovation, we decompose I_1 as

$$\begin{aligned}
I_1 = &- \mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), \frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^k; \xi_m^k)\rangle\Big] \\
&+ \underbrace{\frac{1}{M}\sum_{m\in\mathcal{M}} \mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), \delta_m^k\rangle\Big]}_{:=H_1} \\
= &- \mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), \frac{1}{M}\sum_{m\in\mathcal{M}} \mathbb{E}\left[\nabla \ell(\theta^k; \xi_m^k) \middle| \Theta^k\right]\rangle\Big] + H_1 \\
= &- \mathbb{E}\left[\|\nabla \mathcal{L}(\theta^k)\|^2\right] + H_1
\end{aligned} \tag{4.45}$$

and likewise decompose I_2 as

$$\begin{aligned}
I_2 = &\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \mathcal{L}(\theta^k) + \nabla \mathcal{L}(\theta^k)\Big\|^2\Big] \\
= &\underbrace{\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \mathcal{L}(\theta^k)\Big\|^2\Big]}_{:=H_2} \\
&+ \mathbb{E}\left[\|\nabla \mathcal{L}(\theta^k)\|^2\right] - 2\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), \frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^k; \xi_m^k)\rangle\Big] \\
&+ 2\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), \frac{1}{M}\sum_{m\in\mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})\rangle\Big] \\
= &H_2 + \mathbb{E}\left[\|\nabla \mathcal{L}(\theta^k)\|^2\right] - 2H_1.
\end{aligned} \tag{4.46}$$

We obtain Lemma 11 by plugging (4.45) and (4.46) into (4.44).

4.5.5 Proof of Lemma 12

We next bound H_1 defined in (4.45) separately for different LASG rules. First for LASG-WK1's rule (4.9), we have

$$\begin{aligned}
H_1 \overset{(a)}{=} & \frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k), \tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\rangle\right] \\
& + \frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k), \nabla\ell(\tilde{\theta};\xi^k) - \nabla\ell(\tilde{\theta},\xi_m^{k-\tau_m^k})\rangle\right] \\
\overset{(b)}{\leq} & \frac{L\eta_k}{2}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2\right] + \frac{6DL\eta_k}{M\sqrt{M}}\sum_{m\in\mathcal{M}}\sigma_m^2 \\
& + \sum_{d=1}^{D}\left(\frac{c}{2L\eta_k d_{\max}} + \frac{\sqrt{M}L}{12\eta_k}\right)\mathbb{E}\left[\left\|\Delta^{k-d}\right\|^2\right]
\end{aligned}$$

where (a) is due to the definition of δ_m^k, and (b) is obtained by (4.9), (4.36) with $\varepsilon = \frac{1}{L\eta_k}$, and (4.39) with $\theta^l = \tilde{\theta}$. Note that the definition of $\tilde{\theta}$ in Algorithm 4 implies $l = \lfloor\frac{k}{D}\rfloor \leq k - \tau_m^k$.

For LASG-WK2's rule (4.12), we apply (4.36) with $\varepsilon = \frac{1}{L\eta_k}$ and (4.39) with $l = k - \tau_m^k$, which leads to

$$\begin{aligned}
H_1 = & \frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k), \nabla\ell(\theta^k;\xi_m^k) - \nabla\ell(\theta^{k-\tau_m^k};\xi_m^k))\rangle\right] \\
& + \mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k), \nabla\ell(\theta^{k-\tau_m^k};\xi_m^k) - \nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}))\rangle\right] \\
\leq & \frac{L\eta_k}{2}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] + \frac{6DL\eta_k}{M\sqrt{M}}\sum_{m\in\mathcal{M}}\sigma_m^2 \\
& + \sum_{d=1}^{D}\left(\frac{c}{2L\eta_k d_{\max}} + \frac{\sqrt{M}L}{12\eta_k}\right)\mathbb{E}\left[\|\Delta^{k-d}\|^2\right].
\end{aligned}$$

For LASG-PS's rule (4.14), applying $\mathbb{E}\left[\nabla\ell(\theta^k;\xi_m^k)\,\middle|\,\Theta^k\right]=\nabla\mathcal{L}_m(\theta^k)$, we get

$$\begin{aligned}
H_1 =&\frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),\nabla\mathcal{L}_m(\theta^k)-\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})\rangle\right]\\
&+\frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\Big[\Big\langle\nabla\mathcal{L}(\theta^k),\nabla\mathcal{L}_m(\theta^{k-\tau_m^k})-\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}\rangle\rangle\Big]\\
\overset{(c)}{\leq}&\frac{L\eta_k}{2}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right]+\frac{6DL\eta_k}{2M\sqrt{M}}\sum_{m\in\mathcal{M}}\sigma_m^2\\
&+\sum_{d=1}^{D}\left(\frac{c}{2L\eta_k d_{\max}}+\frac{\sqrt{M}L}{12\eta_k}\right)\mathbb{E}\left[\|\Delta^{k-d}\|^2\right]
\end{aligned}$$

where (c) uses (4.36) with $\varepsilon=\frac{1}{L\eta_k}$ and (4.40) with $l=k-\tau_m^k$.

4.5.6 Proof of Lemma 13

We next bound H_2 defined in (4.46) separately for different LASG rules. For LASG-WK1, using (4.38), we first have

$$\begin{aligned}
H_2 \leq\;& 3\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\tilde{\delta}_m^k-\tilde{\delta}_m^{k-\tau_m^k}\Big\|^2\Big]+3\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^k,\xi_m^k)-\nabla\mathcal{L}(\theta^k))\Big\|^2\Big]\\
&+3\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}(\nabla\ell(\tilde{\theta};\xi_m^k)-\nabla\mathcal{L}_m(\tilde{\theta}))+\frac{1}{M}\sum_{m\in\mathcal{M}}(\nabla\mathcal{L}_m(\tilde{\theta})-\nabla\ell(\tilde{\theta};\xi_m^{k-\tau_m^k}))\Big\|^2\Big]\\
\overset{(a)}{\leq}\;&\frac{3c}{d_{\max}}\sum_{d=1}^{d_{\max}}\mathbb{E}\big[\|\Delta^{k-d}\|^2\big]+\frac{9}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2
\end{aligned}$$

where (a) follows from (4.9), (4.20b), and (4.35).

For LASG-WK2, using (4.38), we have

$$\begin{aligned}
H_2 \leq& 2\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\big(\nabla\ell(\theta^{k-\tau_m^k},\xi_m^{k-\tau_m^k})-\nabla\ell(\theta^k;\xi_m^k)\big)\Big\|^2\Big]\\
&+2\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\big(\nabla\ell(\theta^k;\xi_m^k)-\nabla\mathcal{L}_m(\theta^k)\big)\Big\|^2\Big]\\
\overset{(b)}{\leq}&\frac{2c}{d_{\max}}\sum_{d=1}^{d_{\max}}\mathbb{E}\left[\|\Delta^{k-d}\|^2\right]+\frac{2}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2
\end{aligned}$$

where (b) uses (4.12), (4.20b) and (4.35).

For LASG-PS, using (4.38), we have

$$\begin{aligned}
H_2 \leq & 2\mathbb{E}\Big[\Big\| \sum_{m\in\mathcal{M}} \big(\nabla \ell(\theta^{k-\tau_m^k}, \xi_m^{k-\tau_m^k}) - \nabla \mathcal{L}_m(\theta^{k-\tau_m^k})\big)\Big\|^2\Big] \\
& + 2\mathbb{E}\Big[\Big\| \sum_{m\in\mathcal{M}} \big(\nabla \mathcal{L}_m(\theta^{k-\tau_m^k}) - \nabla \mathcal{L}_m(\theta^k)\big)\Big\|^2\Big] \\
\overset{(c)}{\leq} & \frac{2c}{d_{\max}} \sum_{d=1}^{d_{\max}} \mathbb{E}\left[\|\Delta^{k-d}\|^2\right] + \frac{2}{M^2}\sum_{m\in\mathcal{M}} \sigma_m^2 \\
\leq & \frac{3c}{d_{\max}} \sum_{d=1}^{d_{\max}} \mathbb{E}\|\Delta^{k-d}\|^2 + \frac{9}{M^2}\sum_{m\in\mathcal{M}} \sigma_m^2
\end{aligned}$$

where (c) holds due to (4.14), (4.20b), and (4.35).

4.5.7 Proof of Lemma 14

Plugging Lemmas 12 and 13 into Lemma 11 leads to

$$\begin{aligned}
& \mathbb{E}\left[\mathcal{L}(\theta^{k+1})\right] - \mathbb{E}\left[\mathcal{L}(\theta^k)\right] \\
\leq & -\left(\eta_k - L\eta_k^2 + \frac{L^2\eta_k^3}{2}\right)\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] \\
& + \sum_{d=1}^{D}\left(\left(\frac{3\eta_k}{2d_{\max}} + \frac{1-L\eta_k}{2Ld_{\max}}\right)c + \frac{\sqrt{M}L}{12}\right)\mathbb{E}\left[\|\Delta^{k-d}\|^2\right] \\
& + L\eta_k^2\left(\frac{9}{2} + 6\sqrt{M}D\right)\frac{1}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2
\end{aligned} \tag{4.47}$$

where we use the fact that $L\eta_k \leq 1$.

By definition of $\mathbb{E}[V^k]$, it follows that (with $\gamma_{D+1} = 0$)

$$\begin{aligned}
\mathbb{E}[V^{k+1}] - \mathbb{E}[V^k] = & \mathbb{E}\left[\mathcal{L}(\theta^{k+1})\right] - \mathbb{E}\left[\mathcal{L}(\theta^k)\right] \\
& + \gamma_1\mathbb{E}\left[\|\Delta^k\|^2\right] + \sum_{d=1}^{D}(\gamma_{d+1} - \gamma_d)\mathbb{E}\left[\|\Delta^{k-d}\|^2\right].
\end{aligned}$$

First we decompose $\mathbb{E}\left[\|\Delta^k\|^2\right]$ as

$$\begin{aligned}
\frac{1}{\eta_k^2}\mathbb{E}\left[\|\Delta^k\|^2\right] =& \mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}) - \nabla\mathcal{L}(\theta^k) + \nabla\mathcal{L}(\theta^k)\Big\|^2\Big]\\
\leq& 2\mathbb{E}\big[\|\nabla\mathcal{L}(\theta^k)\|^2\big] + 2\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}) - \nabla\mathcal{L}(\theta^k)\Big\|^2\Big]\\
\overset{(a)}{\leq}& 2\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] + \frac{6c}{d_{\max}}\sum_{d=1}^{D}\mathbb{E}\left[\|\Delta^{k-d}\|^2\right] + \frac{18}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2
\end{aligned}$$

where (a) uses Lemma 13.

Together with (4.47), it follows that

$$\begin{aligned}
\mathbb{E}[V^{k+1}] - \mathbb{E}[V^k] \leq& -\underbrace{\left(\eta_k - (L+2\gamma_1)\eta_k^2\right)}_{:=B_0^k}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right]\\
&+\sum_{d=1}^{D}\underbrace{\bigg(\Big(\eta_k+\frac{1}{2L}\Big)\frac{c}{d_{\max}}+\frac{\sqrt{M}L}{12}+\frac{6c\gamma_1\eta_k^2}{d_{\max}}+\gamma_{d+1}-\gamma_d\bigg)}_{:=A_d^k}\mathbb{E}\Big[\|\Delta^{k-d}\|^2\Big]\\
&+\underbrace{\bigg(\Big(\frac{9}{2}+6\sqrt{M}D\Big)L+18\gamma_1\bigg)}_{:=B_1^k}\frac{\eta_k^2}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2
\end{aligned} \tag{4.48}$$

from which the proof is complete.

4.5.8 Proof of Theorem 4

To ensure $A_d^k \leq 0$ in (4.48) of Lemma 14, it is sufficient to choose $\{\gamma_d\}$ satisfying (with $\gamma_{D+1} = 0$)

$$\Big(\eta_k+\frac{1}{2L}\Big)\frac{c}{d_{\max}}+\frac{\sqrt{M}L}{12}+\frac{6c\gamma_1\eta_k^2}{d_{\max}}+\gamma_{d+1}-\gamma_d \leq 0, \quad 0 \leq d \leq D$$

where the stepsize is chosen as $\eta_k = \eta$, $k = 1, \cdots, K$.

Solve the linear equations above and get

$$\gamma_1 = \frac{(\eta+\frac{1}{2L})cD/d_{\max}+\frac{\sqrt{M}DL}{12}}{1-6cD\eta^2/d_{\max}}. \tag{4.49}$$

Select $c \leq \min\{\frac{d_{\max}}{12D\eta^2}, \frac{d_{\max}\sqrt{M}L^2}{18}\}$ such that $\gamma_1 \leq \frac{\sqrt{M}DL}{3}$. If we further select $\eta \leq \frac{1}{2L+\frac{4}{3}\sqrt{M}DL} \leq \frac{1}{2L+4\gamma_1}$ and then

$$B_0^k = \eta_k - (L + 2\gamma_1)\eta_k^2 \geq \frac{\eta}{2}. \tag{4.50}$$

Summation up (4.33) over $k = 0, \cdots, K-1$, it follows that

$$\sum_{k=0}^{K-1} \frac{\eta_k}{2}\mathbb{E}\Big[\|\nabla\mathcal{L}(\theta^k)\|^2\Big] \leq \mathcal{L}(\theta^0) - \mathcal{L}(\theta^*) + \sum_{k=0}^{K-1}\left(\frac{9}{2} + 12\sqrt{M}D\right)\frac{L\eta_k^2}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2. \tag{4.51}$$

Specifically, if we choose a constant stepsize

$$\eta_k = \eta := \min\left\{\frac{1}{2L + \frac{4}{3}\sqrt{M}DL}, \frac{c_\eta}{\sqrt{K}}\right\} \tag{4.52}$$

where $c_\eta > 0$ is a constant, then

$$\begin{aligned}
&\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] \\
\leq&\frac{2}{K\eta}\left(\mathcal{L}(\theta^0) - \mathcal{L}(\theta^*) + K\left(\frac{9}{2} + 12\sqrt{M}D\right)\frac{L\eta^2}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2\right) \\
\leq&\left(\frac{4L + \frac{8}{3}\sqrt{M}DL}{K} + \frac{2}{c_\eta\sqrt{K}}\right)(\mathcal{L}(\theta^0) - \mathcal{L}(\theta^*)) \\
&+ \frac{c_\eta}{\sqrt{K}}\left(9 + 24\sqrt{M}D\right)\frac{L}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2.
\end{aligned} \tag{4.53}$$

Choosing $c_\eta = \mathcal{O}(M^{\frac{3}{4}}(\sum_{m\in\mathcal{M}}\sigma_m^2)^{-\frac{1}{2}})$ leads to the theorem.

4.5.9 Proof of Theorem 5

Let $\mathbb{E}_Q$ and $\mathbb{E}_{Q,\xi_m}$ denote the expectation with respect to the stochastic quantization Q and both the stochastic quantization Q and the datum ξ_m, respectively.

As a result of [2, Lemma 3.1] and Assumption 4, b-bit quantized gradients have the following unbiasedness property

$$\mathbb{E}_Q\left[Q(\theta;\xi_m)\right] = \nabla\ell(\theta;\xi_m) \tag{4.54}$$

and the bounded variance (with B defined in Assumption 4)

$$\mathbb{E}_{Q,\xi_m}\left[\|Q(\theta;\xi_m)-\nabla\ell(\theta;\xi_m)\|^2\right]\leq\min\left\{\frac{d}{(2^{b-1}-1)^2},\frac{\sqrt{d}}{2^{b-1}-1}\right\}B=:\sigma_Q^2. \tag{4.55}$$

Analogous to the proof of Lemma 11, we can get

$$\begin{aligned}\mathbb{E}\Big[\mathcal{L}(\theta^{k+1})\Big]-\mathbb{E}\Big[\mathcal{L}(\theta^{k})\Big]\leq&-\left(\eta_k-\frac{L\eta_k^2}{2}\right)\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right]\\&+\left(\eta_k-L\eta_k^2\right)H_3+\frac{L\eta_k^2}{2}H_4\end{aligned}$$

where H_3 and H_4 are defined similar to H_1 and H_2 in (4.44).

We first bound H_3 as

$$\begin{aligned}H_3:=&\frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),\nabla\ell(\theta^k;\xi_m^k)-Q(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\rangle\right]\\=&H_1+\frac{1}{M}\sum_{m\in\mathcal{M}}\mathbb{E}\Big[\langle\nabla\mathcal{L}(\theta^k),\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})-Q(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\rangle\!\rangle\Big]\\\overset{(a)}{\leq}&H_1+\frac{\sqrt{M}L}{12\eta_k}\sum_{d=1}^{D}\mathbb{E}\left[\|\Delta^{k-d}\|^2\right]+\frac{6DL\eta_k}{2M\sqrt{M}}\sigma_Q^2\end{aligned} \tag{4.56}$$

where (a) is obtained by steps similar to those of (4.39).

Plugging the bound on H_1 in Lemma 12 into (4.56), we have

$$\begin{aligned}H_3\leq&\frac{L\eta_k}{2}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right]+\sum_{d=1}^{D}\left(\frac{c/d_{\max}}{2L\eta_k}+\frac{\sqrt{M}L}{6\eta_k}\right)\mathbb{E}\left[\|\Delta^{k-d}\|^2\right]\\&+\frac{6DL\eta_k}{\sqrt{M}}\sum_{m\in\mathcal{M}}\left(\sigma_m^2+\frac{\sigma_Q^2}{2}\right).\end{aligned}$$

Likewise, H_4 can be bounded as

$$\begin{aligned}H_4=&\mathbb{E}\Big[\Big\|\nabla\mathcal{L}(\theta^k)-\frac{1}{M}\sum_{m\in\mathcal{M}}Q(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\Big\|^2\Big]\\\overset{(b)}{\leq}&4\mathbb{E}\Big[\Big\|\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})-Q(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k})\Big\|^2\Big]+\frac{4}{3}H_2\\\overset{(c)}{\leq}&\frac{4c}{d_{\max}}\sum_{d=1}^{d_{\max}}\mathbb{E}\|\Delta^{k-d}\|^2+\frac{12}{M^2}\sum_{m\in\mathcal{M}}\left(\sigma_m^2+\frac{\sigma_Q^2}{2}\right)\end{aligned}$$

where (b) uses (4.37) with $\varepsilon = 3$, and (c) uses Lemma 13.

The remaining steps follow those of Theorem 4 with σ_m^2 replaced with $\sigma_m^2 + \frac{\sigma_Q^2}{2}$.

4.5.10 Proof of Theorem 6

Using the PL-condition of $\mathcal{L}(\theta)$, (4.33) can be rewritten as

$$\begin{aligned}\mathbb{E}[V^{k+1}] - \mathbb{E}[V^k] \leq & -2\mu B_0^k \mathbb{E}[\mathcal{L}(\theta^k) - \mathcal{L}(\theta^*)] + B_1^k \sum_{m\in\mathcal{M}} \sigma_m^2 \\ & + \sum_{d=1}^{D} A_d^k \mathbb{E}\Big[\|\Delta^{k-d}\|^2\Big]. \end{aligned} \tag{4.57}$$

If we choose γ_d such that $A_d^k \leq -2\mu B_0^k \gamma_d$ for $d = 1, 2 \ldots, D$, then we have

$$\begin{aligned}\mathbb{E}[V^{k+1}] \leq & (1 - 2\mu B_0^k)\mathbb{E}[V^k] + B_1^k \frac{1}{M^2} \sum_{m\in\mathcal{M}} \sigma_m^2 & (4.58) \\ \leq & \prod_{j=0}^{k} (1 - 2\mu B_0^j) V^0 + \sum_{j=0}^{k} B_1^j \prod_{i=j+1}^{k} \frac{1 - 2\mu B_0^i}{M^2} \sum_{m\in\mathcal{M}} \sigma_m^2. \end{aligned}$$

To ensure $A_d^k \leq -2\mu B_0^k \gamma_d$, note that if $\eta_k \leq \eta \leq \frac{1}{L+2\gamma_1}$, then

$$B_0^k = \eta_k - (L + 2\gamma_1)\eta_k^2 \in [0, \eta_k]. \tag{4.59}$$

Hence, it is sufficient to choose γ_d satisfying ($\gamma_{D+1} = 0$)

$$\Big(\eta_k + \frac{1}{2L}\Big)\frac{c}{d_{\max}} + \frac{\sqrt{M}L}{12} + \frac{6c\gamma_1\eta_k^2}{d_{\max}} + \gamma_{d+1} - \gamma_d \leq -2\mu\eta\gamma_1, \ \forall d.$$

Solve the linear equations above and get

$$\gamma_1 = \frac{(\eta + \frac{1}{2L})cD/d_{\max} + \sqrt{M}DL/12}{1 - 6cD\eta^2/d_{\max} - 2\mu D\eta}. \tag{4.60}$$

Let $\eta_k = \frac{2}{\mu(k+K_0)}$ with $K_0 = \max\{\frac{2(L+\frac{2}{3}\sqrt{M}DL)}{\mu}, 16D\}$, which ensures that

$$\eta_k \leq \eta := \min\left\{\frac{1}{L + 2\gamma_1}, \frac{1}{8\mu D}\right\}. \tag{4.61}$$

Together with the selection $c \leq \min\{\frac{d_{\max}}{24D\eta^2}, \frac{d_{\max}\sqrt{M}L^2}{18}\}$, this ensures that $\gamma_1 \leq \frac{\sqrt{M}DL}{3}$. Plugging into (4.58) leads to

$$\mathbb{E}[V^{k+1}] \leq (1-\mu\eta_k)\mathbb{E}[V^k] + \underbrace{\left(\frac{9}{2} + 12\sqrt{M}D\right)\frac{L}{M^2}\sum_{m\in\mathcal{M}}\sigma_m^2\eta_k^2}_{:=R}.$$

Multiplying over $k = 0, \cdots, K-1$, it follows that

$$\begin{aligned}
\mathbb{E}[V^K] \leq & \prod_{k=0}^{K-1}(1-\mu\eta_k)V^0 + R\sum_{k=0}^{K-1}\eta_k^2\prod_{j=k+1}^{K-1}(1-\mu\eta_j) \\
\leq & \frac{(K_0-2)(K_0-1)}{(K+K_0-2)(K+K_0-1)}V^0 \\
& + \frac{R}{\mu^2}\sum_{k=0}^{K-1}\frac{4}{(k+K_0)^2}\frac{(k+K_0-1)(k+K_0)}{(K+K_0-2)(K+K_0-1)} \\
\leq & \frac{(K_0-1)^2}{(K+K_0-1)^2}V^0 + \frac{4RK}{\mu^2(K+K_0-1)^2}. \qquad (4.62)
\end{aligned}$$

Using the definition of V^0 and the initialization $\theta^{-D} = \cdots = \theta^{-1} = \theta^0$, we complete the proof.

CHAPTER 5

CADA: Communication-Adaptive Distributed Adam

5.1 Introduction

Stochastic gradient descent (SGD) method [93] is prevalent in solving large-scale machine learning problems during the last decades. Although simple to use, the plain-vanilla SGD is often sensitive to the choice of hyper-parameters and sometimes suffer from the slow convergence. Among various efforts to improve SGD, adaptive methods such as AdaGrad [22], Adam [55] and AMSGrad [97] have well-documented empirical performance, especially in training deep neural networks.

In this chapter, we aim to develop a fully adaptive SGD algorithm tailored for the distributed learning. We consider the setting composed of a central server and a set of M workers in $\mathcal{M} := \{1, \ldots, M\}$, where each worker m has its local data ξ_m from a distribution Ξ_m. Workers may have different data distributions $\{\Xi_m\}$, and they collaboratively solve the following problem

$$\min_{\theta \in \mathbb{R}^p} \quad \mathcal{L}(\theta) = \frac{1}{M} \sum_{m \in \mathcal{M}} \mathcal{L}_m(\theta) \quad \text{with} \quad \mathcal{L}_m(\theta) := \mathbb{E}_{\xi_m} \left[\ell(\theta; \xi_m) \right], \; m \in \mathcal{M} \tag{5.1}$$

where $\theta \in \mathbb{R}^p$ is the sought variable and $\{\mathcal{L}_m, m \in \mathcal{M}\}$ are smooth (but not necessarily convex) functions. We focus on the setting where local data ξ_m at each worker m can not be uploaded to the server, and collaboration is needed through communication between the server and workers. This setting often emerges due to the data privacy concerns [73, 49].

To solve (5.1), we can in principle apply the single-node version of the adaptive SGD methods such as Adam [55]: At iteration k, the server broadcasts θ^k to *all* the workers;

each worker m computes $\nabla \ell(\theta^k; \xi_m^k)$ using a randomly selected sample or a minibatch of samples $\{\xi_m^k\} \sim \Xi_m$, and then uploads it to the server; and once receiving stochastic gradients from all workers, the server can simply use the aggregated stochastic gradient $\bar{\boldsymbol{\nabla}}^k = \frac{1}{M}\sum_{m \in \mathcal{M}} \nabla \ell(\theta^k; \xi_m^k)$ to update the parameter via the plain-vanilla single-node Adam. When $\nabla \ell(\theta^k; \xi_m^k)$ is an unbiased gradient of $\mathcal{L}_m(\theta)$, the convergence of this distributed implementation of Adam follows from the original ones [97, 13]. To implement this, however, *all* the workers have to *upload* the fresh $\{\nabla \ell(\theta^k; \xi_m^k)\}$ at each iteration. This prevents the efficient implementation of Adam in scenarios where the communication uplink and downlink are not symmetric, and communication especially upload from workers and the server is costly; e.g., cellular networks [84]. Therefore, *our goal* is to endow an additional dimension of adaptivity to Adam for solving the distributed problem (5.1). In short, on top of its adaptive learning rate and update direction, we want Adam to be communication-adaptive.

5.1.1 Related work

To put our work in context, we review prior contributions that we group in two categories.

5.1.1.1 SGD with adaptive gradients

A variety of SGD variants have been developed recently, including momentum and acceleration [86, 80, 31]. However, these methods are relatively sensitive to the hyper-parameters such as stepsizes, and require significant efforts on finding the optimal parameters.

Adaptive learning rate. One limitation of SGD is that it scales the gradient uniformly in all directions by a pre-determined constant or a sequence of constants (a.k.a. learning rates). This may lead to poor performance when the training data are sparse [22]. To address this issue, adaptive learning rate methods have been developed that scale the gradient in an entry-wise manner by using past gradients, which include AdaGrad [22, 120], AdaDelta [128] and other variants [61]. This simple technique has improved the performance of SGD in some scenarios.

Adaptive SGD. Adaptive SGD methods achieve the best of both worlds, which update the search directions and the learning rates simultaneously using past gradients. Adam [55] and AMSGrad [97] are the representative ones in this category. While these methods are simple-to-use, analyzing their convergence is challenging [97, 113]. Their convergence in the nonconvex setting has been settled only recently [13, 19]. However, most adaptive SGD methods are studied in the single-node setting where data and computation are both centralized. Very recently, adaptive SGD has been studied in the shared memory setting [123], where data is still centralized and communication is not adaptive.

5.1.1.2 Communication-efficient distributed optimization

Popular communication-efficient distributed learning methods belong to two categories: c1) reduce the number of bits per communication round; and, c2) save the number of communication rounds.

For c1), methods are centered around the ideas of *quantization* and *sparsification*.

Reducing communication bits. Quantization has been successfully applied to distributed machine learning. The 1-bit and multi-bits quantization methods have been developed in [98, 2, 110]. More recently, signSGD with majority vote has been developed in [4]. Other advances of quantized gradient schemes include error compensation [122, 53], variance-reduced quantization [129, 38], and quantization to a ternary vector [121, 92]. All of them reduce a significant number of communication bits. Sparsification amounts to transmitting only gradient coordinates with large enough magnitudes exceeding a certain threshold [107, 1]. To avoid losing information of skipping communication, small gradient components will be accumulated and transmitted when they are large enough [66, 105, 3, 119, 45]. Other compression methods also include low-rank approximation [112] and sketching [41]. However, all these methods aim to resolve c1). In some cases, other latencies dominate the bandwidth-dependent transmission latency. This motivates c2).

Reducing communication rounds. One of the most popular techniques in this category is the periodic averaging, e.g., elastic averaging SGD [132], local SGD (a.k.a. FedAvg) [74, 64, 50, 106, 115, 52, 36] or local momentum SGD [125, 116]. In local SGD, workers perform local model updates independently and the models are averaged periodically. Therefore, communication frequency is reduced. However, except [50, 115, 36], most of local SGD methods follow a pre-determined communication schedule that is nonadaptive. Some of them are tailored for the *homogeneous* settings, where the data are independent and identically distributed over all workers. To tackle the heterogeneous case, FedProx has been developed in [60] by solving local subproblems. For learning tasks where the loss function is convex and its conjugate dual is expressible, the dual coordinate ascent-based approaches have been demonstrated to yield impressive empirical performance [42, 70]. Higher-order methods have also been considered [100, 133]. Roughly speaking, algorithms in [60, 42, 70, 100, 133] reduce communication by increasing local gradient computation.

The most related line of work to this chapter is the lazily aggregated gradient (LAG) approach [8, 109]. In contrast to periodic communication, the communication in LAG is adaptive and tailored for the *heterogeneous* settings. Parameters in LAG are updated at the server, and workers only adaptively upload information that is determined to be informative enough. Unfortunately, while LAG has good performance in the deterministic settings (e.g., with full gradient), its performance will be significantly degraded in the stochastic settings. Very recently, FedAvg with local adaptive SGD update has been proposed in [91], which sets a strong benchmark for communication-efficient learning. When the new algorithm in [91] achieves the sweet spot between local SGD and adaptive momentum SGD, the proposed algorithm is very different from ours, and the *averaging period* and the selection of *participating workers* are nonadaptive.

5.1.2 Our approach

We develop a new adaptive SGD algorithm for distributed learning, called **C**ommunication-**A**daptive **D**istributed **A**dam (**CADA**). Akin to the dynamic scaling of every gradient coordinate in Adam, the key motivation of adaptive communication is that during distributed learning, not all communication rounds between the server and workers are equally important. So a natural solution is to use a condition that decides whether the communication is important or not, and then adjust the frequency of communication between a worker and the server. If some workers are not communicating, the server uses their stale information instead of the fresh ones. We will show that this adaptive communication technique can reduce the less informative communication of distributed Adam.

Analogous to the original Adam [55] and its modified version AMSGrad [97], our new CADA approach also uses the exponentially weighted stochastic gradient h^{k+1} as the update direction of θ^{k+1}, and leverages the weighted stochastic gradient magnitude v^{k+1} to inversely scale the update direction h^{k+1}. Different from the direct distributed implementation of Adam that incorporates the fresh (thus unbiased) stochastic gradients $\bar{\boldsymbol{\nabla}}^k = \frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^k;\xi_m^k)$, CADA exponentially combines the aggregated stale stochastic gradients $\boldsymbol{\nabla}^k = \frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\hat{\theta}_m^k;\hat{\xi}_m^k)$, where $\nabla\ell(\hat{\theta}_m^k;\hat{\xi}_m^k)$ is either the fresh stochastic gradient $\nabla\ell(\theta^k;\xi_m^k)$, or an old copy when $\hat{\theta}_m^k \neq \theta^k; \hat{\xi}_m^k \neq \xi_m^k$. Informally, with $\alpha_k > 0$ denoting the stepsize at iteration k, CADA has the following update

$$h^{k+1}=\beta_1 h^k+(1-\beta_1)\boldsymbol{\nabla}^k, \text{ with } \boldsymbol{\nabla}^k=\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\hat{\theta}_m^k;\hat{\xi}_m^k) \tag{5.2a}$$

$$v^{k+1} = \beta_2\hat{v}^k + (1-\beta_2)(\boldsymbol{\nabla}^k)^2 \tag{5.2b}$$

$$\theta^{k+1} = \theta^k - \alpha_k(\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}}h^{k+1} \tag{5.2c}$$

where $\beta_1, \beta_2 > 0$ are the momentum weights, $\hat{V}^{k+1} := \text{diag}(\hat{v}^{k+1})$ is a diagonal matrix whose diagonal vector is $\hat{v}^{k+1} := \max\{v^{k+1}, \hat{v}^k\}$, the constant is $\epsilon > 0$, and I is an identity matrix. To reduce the memory requirement of storing all the stale stochastic gradients $\{\nabla\ell(\theta^k;\xi_m^k)\}$, we can obtain $\boldsymbol{\nabla}^k$ by refining the previous aggregated stochastic gradients $\boldsymbol{\nabla}^{k-1}$ stored in the server via

$$\boldsymbol{\nabla}^k = \boldsymbol{\nabla}^{k-1} + \frac{1}{M}\sum_{m\in\mathcal{M}^k}\delta_m^k \tag{5.3}$$

where $\delta_m^k := \nabla\ell(\theta^k;\xi_m^k) - \nabla\ell(\hat{\theta}_m^k;\hat{\xi}_m^k)$ is the stochastic gradient innovation, and $\mathcal{M}^k$ is the set of workers that upload the stochastic gradient to the server at iteration k. Henceforth, $\hat{\theta}_m^k = \theta^k; \hat{\xi}_m^k = \xi_m^k$, $\forall m \in \mathcal{M}^k$ and $\hat{\theta}_m^k = \hat{\theta}_m^{k-1}; \hat{\xi}_m^k = \hat{\xi}_m^{k-1}$, $\forall m \notin \mathcal{M}^k$. See CADA's implementation in Figure 5.1.

Clearly, the selection of subset $\mathcal{M}^k$ is both critical and challenging. It is critical because it adaptively determines the number of communication rounds per iteration $|\mathcal{M}^k|$. However, it is challenging since 1) the staleness introduced in the Adam update will propagate not only through the momentum gradients but also the adaptive learning rate; 2) the importance of each communication round is dynamic, thus a fixed or nonadaptive condition is ineffective; and 3) the condition needs to be checked efficiently without extra overhead. To overcome these challenges, we develop two adaptive conditions to select $\mathcal{M}^k$ in CADA.

With details deferred to Section 5.2, the contributions of this chapter are listed as follows.

c1) We introduce a novel communication-adaptive distributed Adam (CADA) approach that reuses stale stochastic gradients to reduce communication for distributed implementation of Adam.

c2) We develop a new Lyapunov function to establish convergence of CADA under both the nonconvex and Polyak-Łojasiewicz (PL) conditions even when the datasets are non-i.i.d. across workers. The convergence rate matches that of the original Adam.

c3) We confirm that our novel fully-adaptive CADA algorithms achieve at least 60% performance gains in terms of communication upload over some popular alternatives using numerical tests on logistic regression and neural network training.

5.2 CADA: Communication-Adaptive Distributed Adam

In this section, we develop our communication-adaptive distributed Adam approach. To be more precise in our notations, we henceforth use $\tau_m^k \geq 0$ for the *staleness or age of*

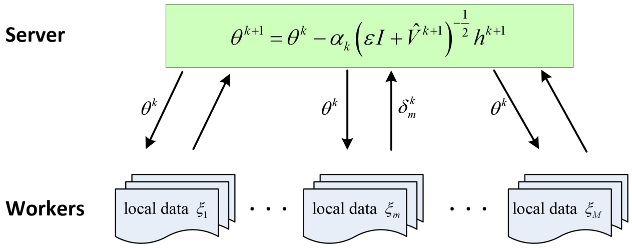

Figure 5.1: The CADA implementation.

information from worker m used by the server at iteration k, e.g., $\hat{\theta}_m^k = \theta^{k-\tau_m^k}$. An age of 0 means "fresh."

5.2.1 Algorithm development of CADA

In this section, we formally develop our CADA method, and present the intuition behind its design.

The key of the CADA design is to *reduce the variance of the innovation measure* in the adaptive condition. We introduce two CADA variants, both of which follow the update (5.2), but they differ in the variance-reduced communication rules.

The first one termed **CADA1** will calculate two stochastic gradient innovations with one $\tilde{\delta}_m^k := \nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\tilde{\theta}; \xi_m^k)$ at the sample ξ_m^k, and one $\tilde{\delta}_m^{k-\tau_m^k} := \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\tilde{\theta}; \xi_m^{k-\tau_m^k})$ at the sample $\xi_m^{k-\tau_m^k}$, where $\tilde{\theta}$ is a snapshot of the previous iterate θ that will be updated every D iterations. As we will show in (5.5), $\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}$ can be viewed as the difference of two variance-reduced gradients calculated at θ^k and $\theta^{k-\tau_m^k}$. Using $\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}$ as the error induced by using stale information, CADA1 will exclude worker m from $\mathcal{M}^k$ if worker m finds

$$\left\| \tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k} \right\|^2 \leq \frac{c}{d_{\max}} \sum_{d=1}^{d_{\max}} \left\| \theta^{k+1-d} - \theta^{k-d} \right\|^2. \tag{5.4}$$

In (5.4), we use the change of parameter θ^k averaged over the past $d_{\max}$ consecutive iterations to measure the progress of algorithm. Intuitively, if (5.4) is satisfied, the error induced by using stale information will not large affect the learning algorithm. In this case, worker m does not upload, and the staleness of information from worker m increases by $\tau_m^{k+1} = \tau_m^k + 1$; otherwise, worker m belongs to $\mathcal{M}^k$, uploads the stochastic gradient innovation δ_m^k, and resets $\tau_m^{k+1} = 1$.

The rationale of CADA1. In contrast to the non-vanishing variance in LAG rule, the CADA1 rule (5.4) reduces its inherent variance. To see this, we can decompose the LHS of (5.4) as the difference of two *variance reduced* stochastic gradients at iteration k and $k - \tau_m^k$. Using the stochastic gradient in SVRG as an example [47], the innovation can be written as

$$\begin{aligned}\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k} = & \big(\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\tilde{\theta}; \xi_m^k) + \nabla \mathcal{L}_m(\tilde{\theta})\big) \\ & - \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\tilde{\theta}; \xi_m^{k-\tau_m^k}) + \nabla \mathcal{L}_m(\tilde{\theta})\Big). \end{aligned} \tag{5.5}$$

Define the minimizer of (5.1) as $\theta^\star$. With derivations given in the supplementary document, the expectation of the LHS of (5.4) can be *upper-bounded* by

$$\mathbb{E}\left[\big\|\tilde{\delta}_m^k - \tilde{\delta}_m^{k-\tau_m^k}\big\|^2\right] = \mathcal{O}\Big(\mathbb{E}[\mathcal{L}(\theta^k)] - \mathcal{L}(\theta^\star) + \mathbb{E}[\mathcal{L}(\theta^{k-\tau_m^k})] - \mathcal{L}(\theta^\star) + \mathbb{E}[\mathcal{L}(\tilde{\theta})] - \mathcal{L}(\theta^\star)\Big). \tag{5.6}$$

If θ^k converges, e.g., $\theta^k, \theta^{k-\tau_m^k}, \tilde{\theta} \to \theta^*$, the RHS of (5.6) diminishes, and thus the LHS of (5.4) diminishes. This is in contrast to the LAG rule *lower-bounded* by a non-vanishing value. Notice that while enjoying the benefit of variance reduction, our communication rule does not need to repeatedly calculate the full gradient $\nabla \mathcal{L}_m(\tilde{\theta})$.

In addition to (5.4), the second rule is termed **CADA2**. The key difference relative to CADA1 is that CADA2 uses $\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta_m^{k-\tau_m^k}; \xi_m^k)$ to estimate the error of using stale information. CADA2 will reuse the stale stochastic gradient $\nabla \ell(\theta_m^{k-\tau_m^k}; \xi_m^{k-\tau_m^k})$ or exclude worker m from $\mathcal{M}^k$ if worker m finds

$$\big\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta_m^{k-\tau_m^k}; \xi_m^k)\big\|^2 \le \frac{c}{d_{\max}} \sum_{d=1}^{d_{\max}} \big\|\theta^{k+1-d} - \theta^{k-d}\big\|^2. \tag{5.7}$$

If (5.7) is satisfied, then worker m does not upload, and the staleness increases by $\tau_m^{k+1} = \tau_m^k + 1$; otherwise, worker m uploads the gradient innovation δ_m^k, and resets the staleness as $\tau_m^{k+1} = 1$.

Algorithm 8 Pseudo-code of CADA; red lines are run only by **CADA1**; blue lines are implemented only by **CADA2**; not both at the same time.

```
**Input:** delay counter $\{\tau_m^0\}$, stepsize $\alpha_k$, constant threshold $c$, max delay $D$.
**for** $k = 0, 1, \ldots, K-1$ **do**
    Server broadcasts $\theta^k$ to all workers.
    All workers set $\tilde{\theta} = \theta^k$ if $k \bmod D = 0$.
    **for** Worker $m = 1, 2, \ldots, M$ **do in parallel**
        Compute $\nabla \ell(\theta^k; \xi_m^k)$ and $\nabla \ell(\tilde{\theta}; \xi_m^k)$.
        Check condition (5.4) with stored $\tilde{\delta}_m^{k-\tau_m^k}$.
        Compute $\nabla \ell(\theta^k; \xi_m^k)$ and $\nabla \ell(\theta_m^{k-\tau_m^k}; \xi_m^k)$.
        Check condition (5.7).
        **if** (5.4) or (5.7) is violated or $\tau_m^k \geq D$ **then**
            Upload $\delta_m^k$.                    ▷ $\tau_m^{k+1} = 1$
        **else**
            Upload nothing.        ▷ $\tau_m^{k+1} = \tau_m^k + 1$
        **end if**
    **end for**
    Server updates $\{h^k, v^k\}$ via (5.2a)-(5.2b).
    Server updates $\theta^k$ via (5.2c).
**end for**
```

The rationale of CADA2. Similar to CADA1, the CADA2 rule (5.7) also reduces its inherent variance, since the LHS of (5.7) can be written as the difference between a *variance reduced* stochastic gradient and a *deterministic* gradient, that is

$$\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) = \Big(\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) + \nabla \mathcal{L}_m(\theta^{k-\tau_m^k}) \Big) \nabla \mathcal{L}_m(\theta^{k-\tau_m^k}). \tag{5.8}$$

With derivations deferred to the supplementary document, similar to (5.6) we also have that $\mathbb{E}[\|\nabla \ell(\theta^k; \xi_m^k) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k)\|^2] \to 0$ as the iterate $\theta^k \to \theta^\star$.

For either (5.4) or (5.7), worker m can check it locally with small memory cost by recursively updating the RHS of (5.4) or (5.7). In addition, worker m will update the stochastic gradient if the staleness satisfies $\tau_m^k \geq D$. We summarize CADA in Algorithm 8.

Computational and memory cost of CADA. In CADA, checking (5.4) and (5.7) will double the computational cost (gradient evaluation) per iteration. Aware of this fact, we have compared the number of iterations and gradient evaluations in simulations (see Figures 5.2-5.5), which will demonstrate that CADA requires *fewer* iterations and also *fewer* gradient queries to achieve a target accuracy. Thus the extra computation is small. In addition, the extra memory for large $d_{\max}$ is low. To compute the RHS of (5.4) or (5.7), each worker only stores the norm of model changes ($d_{\max}$ **scalars**).

5.3 Convergence Analysis of CADA

We present the convergence results of CADA. For all the results, we make some basic assumptions.

Assumption 6 *The loss function $\mathcal{L}(\theta)$ is smooth with the constant L.*

Assumption 7 *Samples $\xi_m^1, \xi_m^2, \ldots$ are independent, and the stochastic gradient $\nabla \ell(\theta; \xi_m^k)$ satisfies $\mathbb{E}_{\xi_m^k}[\nabla \ell(\theta; \xi_m^k)] = \nabla \mathcal{L}_m(\theta)$ and $\|\nabla \ell(\theta; \xi_m^k)\| \leq \sigma_m$.*

Note that Assumptions 6-7 are standard in analyzing Adam and its variants [55, 97, 13, 123].

5.3.1 Key steps of Lyapunov analysis

The convergence results of CADA critically builds on the subsequent Lyapunov analysis. We will start with analyzing the expected descent in terms of $\mathcal{L}(\theta^k)$ by applying one step CADA update.

Lemma 16 *Under Assumptions 6 and 7, if $\alpha_{k+1} \leq \alpha_k$, then $\{\theta^k\}$ generated by CADA satisfy*

$$\mathbb{E}[\mathcal{L}(\theta^{k+1})] - \mathbb{E}[\mathcal{L}(\theta^k)] \leq -\alpha_k(1-\beta_1)\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\boldsymbol{\nabla}^k\rangle\right]$$
$$- \alpha_k\beta_1\mathbb{E}\left[\langle \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\rangle\right] + \left(\frac{L}{2} + \beta_1 L\right)\mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2\right]$$
$$+ \alpha_k(2-\beta_1)\sigma^2\mathbb{E}\Big[\sum_{i=1}^p \Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big] \tag{5.9}$$

where p is the dimension of θ, σ is defined as $\sigma := \frac{1}{M}\sum_{m\in\mathcal{M}}\sigma_m$, and β_1, ϵ are in (5.2).

Lemma 16 contains four terms in the RHS of (5.9): the first two terms quantify the correlations between the gradient direction $\nabla\mathcal{L}(\theta^k)$ and the *stale* stochastic gradient $\boldsymbol{\nabla}^k$ as well as the *state momentum* stochastic gradient h^k; the third term captures the drift of two consecutive iterates; and, the last term estimates the maximum drift of the adaptive stepsizes over $D+1$ iterations.

From Lemma 16, analyzing the progress of $\mathcal{L}(\theta^k)$ under CADA is challenging especially when the effects of staleness and the momentum couple with each other. Because the the state momentum gradient h^k is recursively updated by $\boldsymbol{\nabla}^k$, we will first need the following lemma to characterize the regularity of the stale aggregated stochastic gradients $\boldsymbol{\nabla}^k$, which lays the theoretical foundation for incorporating the properly controlled staleness into the Adam's momentum update.

Lemma 17 *Under Assumptions 6 and 7, if the stepsizes satisfy $\alpha_{k+1} \leq \alpha_k \leq 1/L$, then we have*

$$-\alpha_k\mathbb{E}\Big[\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\boldsymbol{\nabla}^k\rangle\Big] \leq -\frac{\alpha_k}{2}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\right] + \frac{6DL\alpha_k^2\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2$$
$$+ \epsilon^{-\frac{1}{2}}\left(\frac{L}{12} + \frac{c}{2Ld_{\max}}\right)\sum_{d=1}^{D}\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right]. \tag{5.10}$$

Lemma 17 justifies the relevance of the stale yet properly selected stochastic gradients. Intuitively, the first term in the RHS of (5.10) resembles the descent of using SGD with the unbiased stochastic gradient, and the second and third terms will diminish if the stepsizes are diminishing since $\mathbb{E}\left[\|\theta^k - \theta^{k-1}\|^2\right] = \mathcal{O}(\alpha_k^2)$.

In view of Lemmas 16 and 17, we introduce the following **Lyapunov function**:

$$\begin{aligned}\mathcal{V}^k :=& \mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star) - \sum_{j=k}^{\infty} \alpha_j \beta_1^{j-k+1} \left\langle \nabla \mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}} h^k \right\rangle \\ &+ b_k \sum_{d=0}^{D} \sum_{i=1}^{p} (\epsilon + \hat{v}_i^{k-d})^{-\frac{1}{2}} + \sum_{d=1}^{D} \rho_d \|\theta^{k+1-d} - \theta^{k-d}\|^2 \qquad (5.11)\end{aligned}$$

where $\theta^\star$ is the solution of (5.1), $\{b_k\}_{k=1}^K$ and $\{\rho_d\}_{d=1}^D$ are constants specified in the proof.

The design of Lyapunov function in (5.11) is motivated by the progress of $\mathcal{L}(\theta^k)$ in Lemmas 16-17, and also coupled with our communication rules (5.4) and (5.7) that contain the parameter difference term. We find this new Lyapunov function can lead to a much simple proof of Adam and AMSGrad, which is of independent interest. The following lemma captures the progress of the Lyapunov function.

Lemma 18 *Under Assumptions 6-7, if* $\{b_k\}_{k=1}^K$ *and* $\{\rho_d\}_{d=1}^D$ *in* (5.11) *are chosen properly, we have*

$$\mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \leq -\frac{\alpha_k(1-\beta_1)}{2}\left(\epsilon + \frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}} \mathbb{E}\left[\left\|\nabla \mathcal{L}(\theta^k)\right\|^2\right] + \alpha_k^2 C_0 \qquad (5.12)$$

where the constant C_0 *depends on the parameters* $c, \beta_1, \beta_2, \epsilon, D,$ *and* $L, \{\sigma_m^2\}$*.*

The first term in the RHS of (5.12) is strictly negative, and the second term is positive but potentially small since it is $\mathcal{O}(\alpha_k^2)$ with $\alpha_k \to 0$. This implies that the function $\mathcal{V}^k$ will eventually converge if we choose the stepsizes appropriately. Lemma 18 is a generalization of SGD's descent lemma. If we set $\beta_1 = \beta_2 = 0$ in (5.2) and $b_k = 0, \rho_d = 0, \forall d, k$ in (5.11), then Lemma 18 reduces to that of SGD in terms of $\mathcal{L}(\theta^k)$; see e.g., [7, Lemma 4.4].

5.3.2 Main convergence results

Building upon our Lyapunov analysis, we first present the convergence in nonconvex case.

Theorem 7 (nonconvex) *Under Assumptions 6, 7, if we choose* $\alpha_k = \alpha = \mathcal{O}(\frac{1}{\sqrt{K}})$ *and* $\beta_1 < \sqrt{\beta_2} < 1$*, then the iterates* $\{\theta^k\}$ *generated by CADA satisfy*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] = \mathcal{O}\left(\frac{1}{\sqrt{K}}\right). \tag{5.13}$$

From Theorem 7, the convergence rate of CADA in terms of the average gradient norms is $\mathcal{O}(1/\sqrt{K})$, which matches that of the plain-vanilla Adam [97, 13]. Unfortunately, due to the complicated nature of Adam-type analysis, the bound in (5.13) does not achieve the linear speed-up as analyzed for asynchronous nonadaptive SGD such as [63]. However, our analysis is tailored for adaptive SGD and does not make any assumption on the asynchrony, e.g., the set of uploading workers are independent from the past or even independent and identically distributed.

Next we present the convergence results under a slightly stronger assumption.

Assumption 8 *The loss function* $\mathcal{L}(\theta)$ *satisfies the Polyak-Łojasiewicz (PL) condition with the constant* $\mu > 0$*, that is* $\mathcal{L}(\theta) - \mathcal{L}(\theta^*) \leq \frac{1}{2\mu}\left\|\mathcal{L}(\theta)\right\|^2$.

The PL condition is weaker than the strongly convexity, which does not even require convexity [51]. And it is satisfied by a wider range of problems such as least squares for underdetermined linear systems, logistic regression, and also certain types of neural networks.

We next establish the convergence of CADA under this condition.

Theorem 8 (PL-condition) *Under Assumptions 6-8, if we choose the stepsize as* $\alpha_k = \frac{2}{\mu(k+K_0)}$ *for a given constant* K_0*, then* θ^K *generated by Algorithm 8 satisfies*

$$\mathbb{E}\left[\mathcal{L}(\theta^K)\right] - \mathcal{L}(\theta^\star) = \mathcal{O}\left(\frac{1}{K}\right). \tag{5.14}$$

Theorem 8 implies that under the PL-condition of the loss function, the CADA algorithm can achieve the global convergence in terms of the loss function, with a fast rate $\mathcal{O}(1/K)$. Compared with the previous analysis for LAG [8], as we highlighted in Section 5.3.1, the

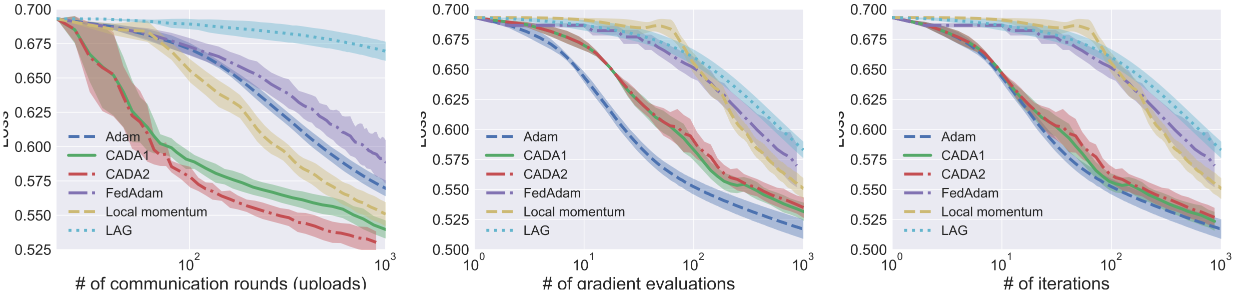

Figure 5.2: Logistic regression loss on *covtype* dataset averaged over 10 Monte Carlo runs.

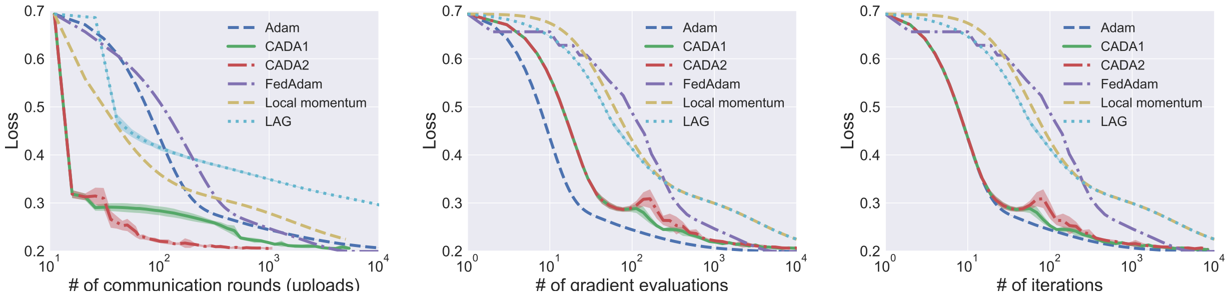

Figure 5.3: Logistic regression loss on *ijcnn1* dataset averaged over 10 Monte Carlo runs.

analysis for CADA is more involved, since it needs to deal with not only the outdated gradients but also the *stochastic momentum* gradients and the *adaptive* matrix learning rates. We tackle this issue by i) considering a new set of communication rules (5.4) and (5.7) with reduced variance; and, ii) incorporating the effect of momentum gradients and the drift of adaptive learning rates in the new Lyapunov function (5.11).

5.4 Numerical Tests

In order to verify our analysis and show the empirical performance of CADA, we conduct experiments in the logistic regression and training neural network tasks, respectively.

In logistic regression, we tested the **covtype** and **ijcnn1** in the main context, and **MNIST** in the appendix. In training neural networks, we tested **MNIST** dataset in the main context, and **CIFAR10** in the appendix. To benchmark CADA, we compared it with

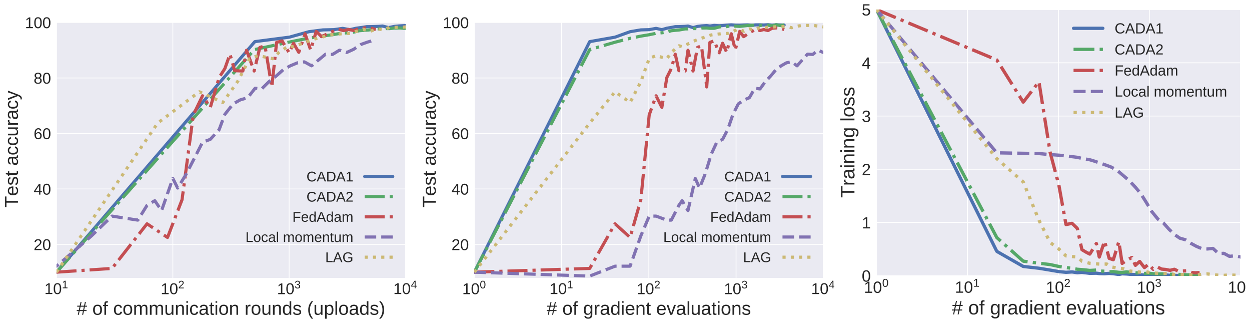

Figure 5.4: Training Neural network for classification on *mnist* dataset.

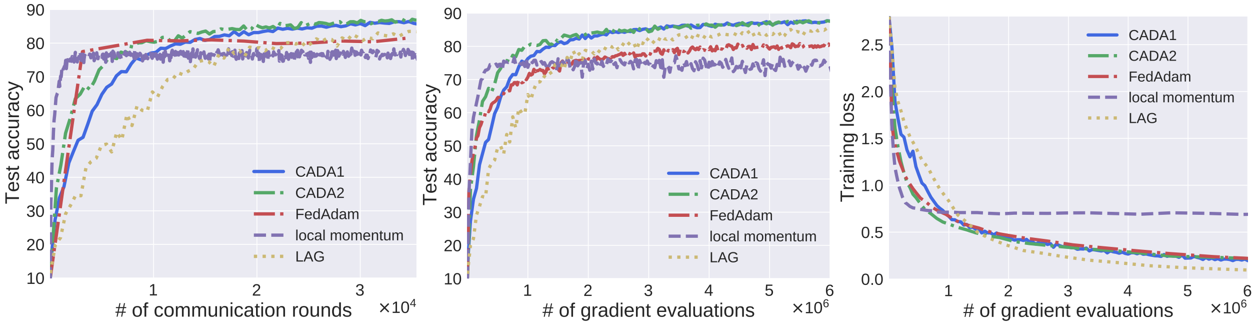

Figure 5.5: Training Neural network for classification on *cifar10* dataset.

some state-of-the-art algorithms, namely ADAM [55], stochastic LAG, local momentum [125, 116] and FedAdam [91]. For local momentum and FedAdam, workers perform model update independently, which are averaged over all workers every H iterations. In simulations, critical parameters are optimized for each algorithm by a grid-search.

All experiments are run on a workstation with an Intel i9-9960x CPU with 128GB memory and four NVIDIA RTX 2080Ti GPUs each with 11GB memory using Python 3.6.

Logistic regression. For CADA, the maximal delay is $D = 100$ and $d_{\max} = 10$. For local momentum and FedAdam, we manually optimize the averaging period as $H = 10$ for *ijcnn1* and $H = 20$ for *covtype*. Results are averaged over 10 Monte Carlo runs.

Tests on logistic regression are reported in Figures 5.2-5.3. In our tests, two CADA variants achieve the similar iteration complexity as the original Adam and outperform all other baselines in most cases. Since our CADA requires two gradient evaluations per iteration, the gradient complexity (e.g., computational complexity) of CADA is higher than Adam, but still smaller than that of other baselines. For logistic regression task, CADA1 and CADA2 save the number of communication uploads by at least one order of magnitude.

Training neural networks. We train a neural network with two convolution-ELU-maxpooling layers followed by two fully-connected layers for 10 classes classification on *mnist*. We use the popular *ResNet20* model on *CIFAR10* dataset, which has 20 and roughly 0.27 million parameters. We searched the best values of H from the grid $\{1, 4, 6, 8, 16\}$ to optimize the testing accuracy vs communication rounds for each algorithm. In CADA, the maximum delay is $D = 50$ and the average interval $d_{\max} = 10$.

Tests on training neural networks are reported in Figures 5.4-5.5. In *mnist*, CADA1 and CADA2 save the number of communication uploads by roughly 60% than local momentum and slightly more than FedAdam. In *cifar10*, CADA1 and CADA2 achieve competitive performance relative to the state-of-the-art algorithms FedAdam and local momentum. We found that if we further enlarge H, FedAdam and local momentum converge fast at the beginning, but reached worse test accuracy (e.g., 5%-15%). It is also evident that the CADA1 and CADA2 rules achieve more communication reduction than the stochastic version of LAG.

5.5 Appendix

We first present some basic inequalities that will be used frequently in this document, and then present the missing derivations of some claims, as well as the proofs of all the lemmas and theorems in the paper, which is followed by details on our experiments. The content of this supplementary document is summarized as follows.

5.5.1 Supporting Lemmas

Define the σ-algebra $\Theta^k = \{\theta^l, 1 \leq l \leq k\}$. For convenience, we also initialize parameters as $\theta^{-D}, \theta^{-D+1}, \ldots, \theta^{-1} = \theta^0$. Some basic facts used in the proof are reviewed as follows.

Fact 1. Assume that $X_1, X_2, \ldots, X_n \in \mathbb{R}^p$ are independent random variables, and $EX_1 = \cdots = EX_n = 0$. Then

$$\mathbb{E}\left[\Big\|\sum_{i=1}^{n} X_i\Big\|^2\right] = \sum_{i=1}^{n} \mathbb{E}\left[\|X_i\|^2\right]. \tag{5.15}$$

Fact 2. (Young's inequality) For any $\theta_1, \theta_2 \in \mathbb{R}^p, \varepsilon > 0$,

$$\langle \theta_1, \theta_2 \rangle \leq \frac{\|\theta_1\|^2}{2\varepsilon} + \frac{\varepsilon\|\theta_2\|^2}{2}. \tag{5.16}$$

As a consequence, we have

$$\|\theta_1 + \theta_2\|^2 \leq \Big(1 + \frac{1}{\varepsilon}\Big)\|\theta_1\|^2 + (1+\varepsilon)\|\theta_2\|^2. \tag{5.17}$$

Fact 3. (Cauchy-Schwarz inequality) For any $\theta_1, \theta_2, \ldots, \theta_n \in \mathbb{R}^p$, we have

$$\Big\|\sum_{i=1}^{n} \theta_i\Big\|^2 \leq n \sum_{i=1}^{n} \|\theta_i\|^2. \tag{5.18}$$

Lemma 19 *For $k - \tau_{\max} \leq l \leq k - D$, if $\{\theta^k\}$ are the iterates generated by CADA, we have*

$$\begin{aligned} &\mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \left(\nabla \ell(\theta^l; \xi_m^k) - \nabla \ell(\theta^l; \xi_m^{k-\tau_m^k})\right)\rangle\right] \\ \leq & \frac{L\epsilon^{-\frac{1}{2}}}{12\alpha_k} \sum_{d=1}^{D} \mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] + 6DL\alpha_k \epsilon^{-\frac{1}{2}} \sigma_m^2 \end{aligned} \tag{5.19}$$

and similarly, we have

$$\begin{aligned} &\mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \left(\nabla \mathcal{L}_m(\theta^l) - \nabla \ell(\theta^l; \theta^{k-\tau_m^k}\right)\rangle\right] \\ \leq & \frac{L\epsilon^{-\frac{1}{2}}}{12\alpha_k} \sum_{d=1}^{D} \mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] + 3DL\alpha_k \epsilon^{-\frac{1}{2}} \sigma_m^2. \end{aligned} \tag{5.20}$$

Proof: We first show the following holds.

$$\begin{aligned} &\mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^l), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \left(\nabla \ell(\theta^l; \xi_m^k) - \nabla \ell(\theta^l; \xi_m^{k-\tau_m^k})\right)\rangle\right] \\ \overset{(a)}{=} & \mathbb{E}\left[\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^l), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \left(\nabla \ell(\theta^l; \xi_m^k) - \nabla \ell(\theta^l; \xi_m^{k-\tau_m^k})\right)\rangle \Big| \Theta^l\Big]\right] \\ \overset{(b)}{=} & \mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^l), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \mathbb{E}\Big[\nabla \ell(\theta^l; \xi_m^k) - \nabla \ell(\theta^l; \xi_m^{k-\tau_m^k}) \big| \Theta^l\Big]\rangle\right] \\ = & \mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^l), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \left(\nabla \mathcal{L}_m(\theta^l) - \nabla \mathcal{L}_m(\theta^l)\right)\rangle\right] = 0 \end{aligned} \tag{5.21}$$

where (a) follows from the law of total probability, and (b) holds because $\hat{V}^{k-D}$ is deterministic conditioned on Θ^l when $k-D \leq l$.

We first prove (5.19) by decomposing it as

$$\begin{aligned}
&\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\left(\nabla\ell(\theta^l;\xi_m^k)-\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\right)\rangle\right]\\
\overset{(c)}{=}&\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k)-\nabla\mathcal{L}(\theta^l),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\left(\nabla\ell(\theta^l;\xi_m^k)-\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\right)\rangle\right]\\
\overset{(d)}{\leq}&L\mathbb{E}\left[\left\|(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{4}}\right\|\left\|\theta^k-\theta^l\right\|\left\|(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{4}}\left(\nabla\ell(\theta^l;\xi_m^k)-\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\right)\right\|\right]\\
\overset{(e)}{\leq}&\frac{L\epsilon^{-\frac{1}{2}}}{12D\alpha_k}\underbrace{\mathbb{E}\left[\|\theta^k-\theta^l\|^2\right]}_{I_1}+\frac{6DL\alpha_k\epsilon^{-\frac{1}{2}}}{2}\underbrace{\mathbb{E}\left[\|\nabla\ell(\theta^l;\xi_m^k)-\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\|^2\right]}_{I_2}
\end{aligned}\tag{5.22}$$

where (c) holds due to (5.21), (d) uses Assumption 6, and (e) applies the Young's inequality.

Applying the Cauchy-Schwarz inequality to I_1, we have

$$\begin{aligned}
I_1=&\mathbb{E}\Big[\Big\|\sum_{d=1}^{k-l}(\theta^{k+1-d}-\theta^{k-d})\Big\|^2\Big]\\
\leq&(k-l)\sum_{d=1}^{k-l}\mathbb{E}\Big[\|\theta^{k+1-d}-\theta^{k-d}\|^2\Big]\leq D\sum_{d=1}^{D}\mathbb{E}\Big[\|\theta^{k+1-d}-\theta^{k-d}\|^2\Big].
\end{aligned}\tag{5.23}$$

Applying Assumption 7 to I_2, we have

$$\begin{aligned}
I_2=&\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^k)-\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big\|^2\Big]\\
=&\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^k)\big\|^2\Big]+\mathbb{E}\Big[\big\|\nabla\ell(\theta^l;\xi_m^{k-\tau_m^k})\big\|^2\Big]\leq 2\sigma_m^2
\end{aligned}\tag{5.24}$$

where the last inequality uses Assumption 7. Plugging (5.23) and (5.24) into (5.22), it leads to (5.19). Likewise, following the steps to (5.22), it can be verified that (5.20) also holds true.

Lemma 20 *Under Assumption 7, the parameters $\{h^k,\hat{v}^k\}$ of CADA in Algorithm 8 satisfy*

$$\|h^k\|\leq\sigma,\quad\forall k;\quad\hat{v}_i^k\leq\sigma^2,\quad\forall k,i\tag{5.25}$$

where $\sigma:=\frac{1}{M}\sum_{m\in\mathcal{M}}\sigma_m$.

Proof: Using Assumption 2, it follows that

$$\|\boldsymbol{\nabla}^k\| = \left\| \frac{1}{M} \sum_{m \in \mathcal{M}} \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) \right\| \leq \frac{1}{M} \sum_{m \in \mathcal{M}} \left\| \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) \right\| \leq \frac{1}{M} \sum_{m \in \mathcal{M}} \sigma_m = \sigma. \tag{5.26}$$

Therefore, from the update (5.2a), we have

$$\|h^{k+1}\| \leq \beta_1 \|h^k\| + (1-\beta_1)\|\boldsymbol{\nabla}^k\| \leq \beta_1 \|h^k\| + (1-\beta_1)\sigma.$$

Since $\|h^1\| \leq \sigma$, if follows by induction that $\|h^{k+1}\| \leq \sigma,\ \forall k$.

Using Assumption 2, it follows that

$$\begin{aligned}
(\nabla_i^k)^2 &= \left(\frac{1}{M} \sum_{m \in \mathcal{M}} \nabla_i \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) \right)^2 \\
&\leq \frac{1}{M} \sum_{m \in \mathcal{M}} \left(\nabla_i \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) \right)^2 \\
&\leq \frac{1}{M} \sum_{m \in \mathcal{M}} \left\| \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) \right\|^2 = \frac{1}{M} \sum_{m \in \mathcal{M}} \sigma_m^2 \leq \sigma^2.
\end{aligned} \tag{5.27}$$

Similarly, from the update (5.2b), we have

$$\hat{v}_i^{k+1} \leq \max\{\hat{v}_i^k, \beta_2 \hat{v}_i^k + (1-\beta_2)(\nabla_i^k)^2\} \leq \max\{\hat{v}_i^k, \beta_2 \hat{v}_i^k + (1-\beta_2)\sigma^2\}.$$

Since $v_i^1 = \hat{v}_i^1 \leq \sigma^2$, if follows by induction that $\hat{v}_i^{k+1} \leq \sigma^2$.

Lemma 21 *Under Assumption 7, the iterates $\{\theta^k\}$ of CADA in Algorithm 8 satisfy*

$$\left\| \theta^{k+1} - \theta^k \right\|^2 \leq \alpha_k^2 p (1-\beta_2)^{-1} (1-\beta_3)^{-1} \tag{5.28}$$

where p is the dimension of θ, $\beta_1 < \sqrt{\beta_2} < 1$, and $\beta_3 := \beta_1^2/\beta_2$.

Proof: Choosing $\beta_1 < 1$ and defining $\beta_3 := \beta_1^2/\beta_2$, it can be verified that

$$\begin{aligned}
|h_i^{k+1}| &= \left|\beta_1 h_i^k + (1-\beta_1)\nabla_i^k\right| \beta_1 |h_i^k| + |\nabla_i^k| \\
&\leq \beta_1 \left(\beta_1 |h_i^{k-1}| + |\nabla_i^{k-1}|\right) + |\nabla_i^k| \\
&\leq \sum_{l=0}^{k} \beta_1^{k-l} |\nabla_i^l| = \sum_{l=0}^{k} \sqrt{\beta_3}^{k-l} \sqrt{\beta_2}^{k-l} |\nabla_i^l| \\
&\overset{(a)}{\leq} \left(\sum_{l=0}^{k} \beta_3^{k-l}\right)^{\frac{1}{2}} \left(\sum_{l=0}^{k} \beta_2^{k-l} (\nabla_i^l)^2\right)^{\frac{1}{2}} \\
&\leq (1-\beta_3)^{-\frac{1}{2}} \left(\sum_{l=0}^{k} \beta_2^{k-l} (\nabla_i^l)^2\right)^{\frac{1}{2}}
\end{aligned} \tag{5.29}$$

where (a) follows from the Cauchy-Schwartz inequality.

For $\hat{v}_i^k$, first we have that $\hat{v}_i^1 \geq (1-\beta_2)(\nabla_i^1)^2$. Then since

$$\hat{v}_i^{k+1} \geq \beta_2 \hat{v}_i^k + (1-\beta_2)(\nabla_i^k)^2$$

by induction we have

$$\hat{v}_i^{k+1} \geq (1-\beta_2) \sum_{l=0}^{k} \beta_2^{k-l} (\nabla_i^l)^2. \tag{5.30}$$

Using (5.29) and (5.30), we have

$$\begin{aligned}
|h_i^{k+1}|^2 \leq & (1-\beta_3)^{-1} \left(\sum_{l=0}^{k} \beta_2^{k-l} (\nabla_i^l)^2\right) \\
\leq & (1-\beta_2)^{-1} (1-\beta_3)^{-1} \hat{v}_i^{k+1}.
\end{aligned}$$

From the update (5.2c), we have

$$\begin{aligned}
\|\theta^{k+1} - \theta^k\|^2 &= \alpha_k^2 \sum_{i=1}^{p} \left(\epsilon + \hat{v}_i^{k+1}\right)^{-1} |h_i^{k+1}|^2 \\
&\leq \alpha_k^2 p (1-\beta_2)^{-1} (1-\beta_3)^{-1}
\end{aligned} \tag{5.31}$$

which completes the proof.

5.5.2 Proof of Lemma 16

Using the smoothness of $\mathcal{L}(\theta)$ in Assumption 6, we have

$$\begin{aligned}\mathcal{L}(\theta^{k+1}) \leq& \mathcal{L}(\theta^k) + \langle \nabla\mathcal{L}(\theta^k), \theta^{k+1} - \theta^k\rangle + \frac{L}{2}\left\|\theta^{k+1} - \theta^k\right\|^2\\ =& \mathcal{L}(\theta^k) - \alpha_k\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} h^{k+1}\rangle + \frac{L}{2}\left\|\theta^{k+1} - \theta^k\right\|^2. \end{aligned} \tag{5.32}$$

We can further decompose the inner product as

$$\begin{aligned}&- \langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} h^{k+1}\rangle\\ =& - (1-\beta_1)\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} \boldsymbol{\nabla}^k\rangle \underbrace{-\beta_1\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^k\rangle}_{I_1^k}\\ &\underbrace{-\langle \nabla\mathcal{L}(\theta^k), \left((\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} - (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}}\right) h^{k+1}\rangle}_{I_2^k} \end{aligned} \tag{5.33}$$

where we again decompose the first inner product as

$$\begin{aligned}-(1-\beta_1)\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} \boldsymbol{\nabla}^k\rangle =& \underbrace{-(1-\beta_1)\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \boldsymbol{\nabla}^k\rangle}_{I_3^k}\\ &\underbrace{-(1-\beta_1)\langle \nabla\mathcal{L}(\theta^k), \left((\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} - (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\right) \boldsymbol{\nabla}^k\rangle}_{I_4^k}. \end{aligned} \tag{5.34}$$

Next, we bound the terms $I_1^k, I_2^k, I_3^k, I_4^k$ separately.

Taking expectation on I_1^k conditioned on Θ^k, we have

$$\begin{aligned}\mathbb{E}[I_1^k \mid \Theta^k] =& -\mathbb{E}\left[\beta_1\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^k\rangle \mid \Theta^k\right]\\ =& -\beta_1\langle \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^k\rangle - \beta_1\langle \nabla\mathcal{L}(\theta^k) - \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^k\rangle\\ \overset{(a)}{\leq}& -\beta_1\langle \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^k\rangle + \alpha_{k-1}^{-1}\beta_1 L\left\|\theta^k - \theta^{k-1}\right\|^2\\ \overset{(b)}{\leq}& \beta_1\left(I_1^{k-1} + I_2^{k-1} + I_3^{k-1} + I_4^{k-1}\right) + \alpha_{k-1}^{-1}\beta_1 L\left\|\theta^k - \theta^{k-1}\right\|^2 \end{aligned} \tag{5.35}$$

where follows from the L-smoothness of $\mathcal{L}(\theta)$ implied by Assumption 6; and (b) uses again the decomposition (5.33) and (5.34).

Taking expectation on I_2^k over all the randomness, we have

$$\begin{aligned}\mathbb{E}[I_2^k] =&\mathbb{E}\Big[-\langle\nabla\mathcal{L}(\theta^k), \Big((\epsilon I+\hat{V}^{k+1})^{-\frac{1}{2}}-(\epsilon I+\hat{V}^k)^{-\frac{1}{2}}\Big)\, h^{k+1}\rangle\Big]\\ =&\mathbb{E}\Big[\sum_{i=1}^p \nabla_i\mathcal{L}(\theta^k)h_i^{k+1}\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big]\\ \overset{(d)}{\leq}&\mathbb{E}\Big[\|\nabla\mathcal{L}(\theta^k)\|\|h^{k+1}\|\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big]\\ \overset{(e)}{\leq}&\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big] \end{aligned} \tag{5.36}$$

where (d) follows from the Cauchy-Schwarz inequality and (e) is due to Assumption 7. Regarding I_3^k, we will bound separately in Lemma 17.

Taking expectation on I_4^k over all the randomness, we have

$$\begin{aligned}\mathbb{E}[I_4^k] =&\mathbb{E}\Big[-(1-\beta_1)\langle\nabla\mathcal{L}(\theta^k), \Big((\epsilon I+\hat{V}^k)^{-\frac{1}{2}}-(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\Big)\,\boldsymbol{\nabla}^k\rangle\Big]\\ =&-(1-\beta_1)\mathbb{E}\Big[\sum_{i=1}^p \nabla_i\mathcal{L}(\theta^k)\boldsymbol{\nabla}_i^k\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}\Big)\Big]\\ \leq&(1-\beta_1)\mathbb{E}\Big[\|\nabla\mathcal{L}(\theta^k)\|\|\boldsymbol{\nabla}^k\|\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}\Big)\Big]\\ \leq&(1-\beta_1)\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}\Big)\Big]. \end{aligned} \tag{5.37}$$

Taking expectation on (5.32) over all the randomness, and plugging (5.35), (5.36), and

(5.37), we have

$$
\begin{aligned}
\mathbb{E}[\mathcal{L}(\theta^{k+1})]-\mathbb{E}[\mathcal{L}(\theta^k)]\leq& -\alpha_k\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k+1})^{-\frac{1}{2}}h^{k+1}\rangle\right]+\frac{L}{2}\mathbb{E}\left[\left\|\theta^{k+1}-\theta^k\right\|^2\right]\\
=&\alpha_k\mathbb{E}\left[I_1^k+I_2^k+I_3^k+I_4^k\right]+\frac{L}{2}\mathbb{E}\left[\left\|\theta^{k+1}-\theta^k\right\|^2\right]\\
\leq&-\alpha_k(1-\beta_1)\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\boldsymbol{\nabla}^k\rangle\right]\\
&-\alpha_k\beta_1\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^{k-1}),(\epsilon I+\hat{V}^{k})^{-\frac{1}{2}}h^k\rangle\right]\\
&+\alpha_k\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\left((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\Big]\\
&+\alpha_k(1-\beta_1)\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\left((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k})^{-\frac{1}{2}}\right)\Big]\\
&+\left(\frac{L}{2}+\alpha_k\alpha_{k-1}^{-1}\beta_1L\right)\mathbb{E}\left[\|\theta^{k+1}-\theta^k\|^2\right].
\end{aligned}\tag{5.38}
$$

Since $(\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}\leq(\epsilon+\hat{v}_i^{k-1})^{-\frac{1}{2}}$, we have

$$
\begin{aligned}
&\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^k)^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)+(1-\beta_1)\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k})^{-\frac{1}{2}}\Big)\Big]\\
\leq&(2-\beta_1)\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\left((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}-(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\Big].
\end{aligned}\tag{5.39}
$$

Plugging (5.39) into (5.38) leads to the statement of Lemma 16.

5.5.3 Proof of Lemma 17

We first analyze the inner produce under CADA2 and then CADA1.

First recall that $\bar{\boldsymbol{\nabla}}^k=\frac{1}{M}\sum_{m\in\mathcal{M}}\nabla\ell(\theta^k;\xi_m^k)$. Using the law of total probability implies that

$$
\begin{aligned}
\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\bar{\boldsymbol{\nabla}}^k\rangle\right]=&\,\mathbb{E}\left[\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\bar{\boldsymbol{\nabla}}^k\rangle\mid\Theta^k\right]\right]\\
=&\,\mathbb{E}\left[\langle\nabla\mathcal{L}(\theta^k),(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}\mathbb{E}\left[\bar{\boldsymbol{\nabla}}^k\mid\Theta^k\right]\rangle\right]\\
=&\,\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\right].
\end{aligned}\tag{5.40}
$$

Taking expectation on $\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \boldsymbol{\nabla}^k \rangle$ over all randomness, we have

$$
\begin{aligned}
&-\mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \boldsymbol{\nabla}^k \rangle\right] \\
=&-\mathbb{E}\left[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \bar{\boldsymbol{\nabla}}^k \rangle\right] \\
&-\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \frac{1}{M} \sum_{m \in \mathcal{M}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\theta^k; \xi_m^k)\Big)\rangle\Big] \\
\overset{(a)}{=}&-\mathbb{E}\left[\left\|\nabla \mathcal{L}(\theta^k)\right\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\right] \\
&-\frac{1}{M} \sum_{m \in \mathcal{M}} \mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\theta^k; \xi_m^k)\Big)\rangle\Big] \qquad (5.41)
\end{aligned}
$$

where (a) uses (5.40).

Decomposing the inner product, for the CADA2 rule (5.7), we have

$$
\begin{aligned}
&-\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\theta^k; \xi_m^k)\Big)\rangle\Big] \\
=&-\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^{k-\tau_m^k}) - \nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k)\Big)\rangle\Big] \\
&-\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) - \nabla \ell(\theta^k; \xi_m^k)\Big)\rangle\Big] \\
\overset{(b)}{\leq}&\frac{L\epsilon^{-\frac{1}{2}}}{12\alpha_k} \sum_{d=1}^{D} \mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] + 6DL\alpha_k \epsilon^{-\frac{1}{2}} \sigma_m^2 \\
&-\mathbb{E}\Big[\Big\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) - \nabla \ell(\theta^k; \xi_m^k)\Big)\Big\rangle\Big] \qquad (5.42)
\end{aligned}
$$

where (b) follows from Lemma 19.

Using the Young's inequality, we can bound the last inner product in (5.42) as

$$
\begin{aligned}
&-\mathbb{E}\Big[\langle \nabla \mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}} \Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) - \nabla \ell(\theta^k; \xi_m^k)\Big)\rangle\Big] \\
\leq&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla \mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{1}{2}\mathbb{E}\Big[\Big\|(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\Big\|\Big\|\Big(\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) - \nabla \ell(\theta^k; \xi_m^k)\Big)\Big\|^2\Big] \\
\overset{(g)}{\leq}&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla \mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{1}{2}\mathbb{E}\Big[\Big\|(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\Big\|\Big\|\nabla \ell(\theta^{k-\tau_m^k}; \xi_m^k) - \nabla \ell(\theta^k; \xi_m^k)\Big\|^2\Big] \\
\overset{(h)}{\leq}&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla \mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{c}{2d_{\max}}\mathbb{E}\Big[\Big\|(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\Big\| \sum_{d=1}^{d_{\max}} \left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2\Big] \\
\overset{(i)}{\leq}&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla \mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{c\epsilon^{-\frac{1}{2}}}{2d_{\max}} \sum_{d=1}^{D} \mathbb{E}\Big[\left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2\Big] \qquad (5.43)
\end{aligned}
$$

where (g) follows from the Cauchy-Schwarz inequality, and (h) uses the adaptive communication condition (5.7) in CADA2, and (i) follows since $\hat{V}^{k-D}$ is entry-wise nonnegative and $\left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2$ is nonnegative.

Similarly for CADA1's condition (5.4), we have

$$
\begin{aligned}
&-\mathbb{E}\Big[\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\left(\nabla\ell(\theta^{k-\tau_m^k};\xi_m^{k-\tau_m^k}) - \nabla\ell(\theta^k;\xi_m^k)\right)\rangle\Big]\\
=&-\mathbb{E}\Big[\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\left(\nabla\ell(\tilde{\theta};\xi_m^{k-\tau_m^k}) - \nabla\ell(\tilde{\theta};\xi_m^k)\right)\rangle\Big]\\
&-\mathbb{E}\Big[\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\left(\tilde{\delta}_m^{k-\tau_m^k} - \tilde{\delta}_m^k)\right)\rangle\Big]\\
\overset{(j)}{\leq}&\frac{L\epsilon^{-\frac{1}{2}}}{12\alpha_k}\sum_{d=1}^{D}\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] + 6DL\alpha_k\epsilon^{-\frac{1}{2}}\sigma_m^2 - \mathbb{E}\Big[\langle\nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\Big(\tilde{\delta}_m^{k-\tau_m^k} - \tilde{\delta}_m^k\Big)\rangle\Big]
\end{aligned}
\tag{5.44}
$$

where (j) follows from Lemma 19 since $\tilde{\theta}$ is a snapshot among $\{\theta^k, \cdots, \theta^{k-D}\}$.

And the last product in (5.44) is bounded by

$$
\begin{aligned}
&-\mathbb{E}\Big[\langle\nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}\Big(\tilde{\delta}_m^{k-\tau_m^k} - \tilde{\delta}_m^k\Big)\rangle\Big]\\
\leq&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla\mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{c}{2}\mathbb{E}\Big[\Big\|(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2d_{\max}}}\Big\|\sum_{d=1}^{d_{\max}}\left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2\Big]\\
\overset{(i)}{\leq}&\frac{1}{2}\mathbb{E}\Big[\Big\|\nabla\mathcal{L}(\theta^k)\Big\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\Big] + \frac{c\epsilon^{-\frac{1}{2}}}{2d_{\max}}\sum_{d=1}^{D}\mathbb{E}\Big[\left\|\theta^{k+1-d} - \theta^{k-d}\right\|^2\Big].
\end{aligned}
\tag{5.45}
$$

Combining (5.41)-(5.45) leads to the desired statement for CADA1 and CADA2.

5.5.4 Proof of Lemma 18

For notational brevity, we re-write the Lyapunov function (5.11) as

$$
\begin{aligned}
\mathcal{V}^k := \mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star) - c_k\left\langle\nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\right\rangle&\\
+ b_k\sum_{d=0}^{D}\sum_{i=1}^{p}(\epsilon + \hat{v}_i^{k-d})^{-\frac{1}{2}} + \sum_{d=1}^{D}\rho_d\|\theta^{k+1-d} - \theta^{k-d}\|^2&
\end{aligned}
\tag{5.46}
$$

where $\{c_k\}$ are some positive constants.

Therefore, taking expectation on the difference of $\mathcal{V}^k$ and $\mathcal{V}^{k+1}$ in (5.46), we have (with $\rho_{D+1} = 0$)

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] =& \mathbb{E}[\mathcal{L}(\theta^{k+1})] - \mathbb{E}[\mathcal{L}(\theta^k)] - c_{k+1}\mathbb{E}\left[\left\langle \nabla\mathcal{L}(\theta^k), (\epsilon I + \hat{V}^{k+1})^{-\frac{1}{2}} h^{k+1}\right\rangle\right] \\
&+ c_k \mathbb{E}\left[\left\langle \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^{k})^{-\frac{1}{2}} h^{k}\right\rangle\right] \\
&+ b_{k+1}\sum_{d=0}^{D}\sum_{i=1}^{p}(\epsilon + \hat{v}_i^{k+1-d})^{-\frac{1}{2}} - b_k \sum_{d=0}^{D}\sum_{i=1}^{p}(\epsilon + \hat{v}_i^{k-d})^{-\frac{1}{2}} \\
&+ \rho_1 \mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2\right] + \sum_{d=1}^{D}(\rho_{d+1} - \rho_d)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] \\
\overset{(a)}{\leq}& (\alpha_k + c_{k+1})\mathbb{E}\left[I_1^k + I_2^k + I_3^k + I_4^k\right] - c_k \mathbb{E}\left[I_1^{k-1} + I_2^{k-1} + I_3^{k-1} + I_4^{k-1}\right] \\
&+ b_{k+1}\sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k+1})^{-\frac{1}{2}}\Big] - b_k \sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k-D})^{-\frac{1}{2}}\Big] \\
&+ \sum_{d=1}^{D}(b_{k+1} - b_k)\sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k+1-d})^{-\frac{1}{2}}\Big] + \left(\frac{L}{2} + \rho_1\right)\mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2\right] \\
&+ \sum_{d=1}^{D}(\rho_{d+1} - \rho_d)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right]
\end{aligned} \tag{5.47}$$

where (a) uses the smoothness in Assumption 6 and the definition of $I_1^k, I_2^k, I_3^k, I_4^k$ in (5.33) and (5.34).

Note that we can bound $(\alpha_k + c_{k+1})\mathbb{E}\left[I_1^k + I_2^k + I_3^k + I_4^k\right]$ the same as (5.33) in the proof of Lemma 16. In addition, Lemma 17 implies that

$$\begin{aligned}
\mathbb{E}[I_3^k] \leq& -\frac{1-\beta_1}{2}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2_{(\epsilon I + \hat{V}^{k-D})^{-\frac{1}{2}}}\right] \\
&+ (1-\beta_1)\epsilon^{-\frac{1}{2}}\left(\frac{L}{12\alpha_k} + \frac{c}{2d_{\max}}\right)\sum_{d=1}^{D}\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] + (1-\beta_1)\frac{6DL\alpha_k\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2.
\end{aligned} \tag{5.48}$$

Hence, plugging Lemma 16 with α_k replaced by $\alpha_k + c_{k+1}$ into (5.47), together with (5.48), leads to

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \leq & -(\alpha_k + c_{k+1})\left(\frac{1-\beta_1}{2}\right)\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\right] \\
& + (\alpha_k + c_{k+1})(1-\beta_1)\epsilon^{-\frac{1}{2}}\left(\frac{L}{12\alpha_k} + \frac{c}{2d_{\max}}\right)\sum_{d=1}^{D}\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] \\
& + (\alpha_k + c_{k+1})(1-\beta_1)\frac{6DL\alpha_k\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 \\
& + ((\alpha_k + c_{k+1})\beta_1 - c_k)\mathbb{E}\left[I_1^{k-1} + I_2^{k-1} + I_3^{k-1} + I_4^{k-1}\right] \\
& + (\alpha_k + c_{k+1})(2-\beta_1)\sigma^2\mathbb{E}\Big[\sum_{i=1}^{p}\left((\epsilon + \hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon + \hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\Big] \\
& + b_{k+1}\sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k+1})^{-\frac{1}{2}}\Big] - b_k\sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k-D})^{-\frac{1}{2}}\Big] \\
& + \sum_{d=1}^{D}(b_{k+1} - b_k)\sum_{i=1}^{p}\mathbb{E}\Big[(\epsilon + \hat{v}_i^{k+1-d})^{-\frac{1}{2}}\Big] + \sum_{d=1}^{D}(\rho_{d+1} - \rho_d)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right] \\
& + \left(\frac{L}{2} + \rho_1 + (\alpha_k + c_{k+1})\alpha_{k-1}^{-1}\beta_1 L\right)\mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2\right]. \qquad (5.49)
\end{aligned}$$

Select $\alpha_k \leq \alpha_{k-1}$ and $c_k := \sum_{j=k}^{\infty}\alpha_j\beta_1^{j-k+1} \leq (1-\beta_1)^{-1}\alpha_k$ so that $(\alpha_k + c_{k+1})\beta_1 = c_k$ and

$$\begin{aligned}
(\alpha_k + c_{k+1})(1-\beta_1) &\leq (\alpha_k + (1-\beta_1)^{-1}\alpha_{k+1})(1-\beta_1) \\
&\leq \alpha_k(1 + (1-\beta_1)^{-1})(1-\beta_1) = \alpha_k(2-\beta_1).
\end{aligned}$$

In addition, select b_k to ensure that $b_{k+1} \leq b_k$. Then it follows from (5.49) that

$$\begin{aligned}\mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \leq & -\frac{\alpha_k(1-\beta_1)}{2}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2_{(\epsilon I+\hat{V}^{k-D})^{-\frac{1}{2}}}\right] + (2-\beta_1)\alpha_k^2\frac{6DL\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 \\ & + (2-\beta_1)\alpha_k\epsilon^{-\frac{1}{2}}\left(\frac{L}{12\alpha_k}+\frac{c}{2d_{\max}}\right)\sum_{d=1}^{D}\mathbb{E}\left[\|\theta^{k+1-d}-\theta^{k-d}\|^2\right] \\ & + \left(\frac{(2-\beta_1)^2}{(1-\beta_1)}\alpha_k\sigma^2 - b_k\right)\mathbb{E}\Big[\sum_{i=1}^{p}\left((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\Big] \\ & + \left(\frac{L}{2}+\rho_1+(1-\beta_1)^{-1}L\right)\mathbb{E}\left[\|\theta^{k+1}-\theta^k\|^2\right] \\ & + \sum_{d=1}^{D}(\rho_{d+1}-\rho_d)\mathbb{E}\left[\|\theta^{k+1-d}-\theta^{k-d}\|^2\right] \end{aligned} \tag{5.50}$$

where we have also used the fact that $-(\alpha_k + c_{k+1})\left(\frac{1-\beta_1}{2}\right) \leq -\frac{\alpha_k(1-\beta_1)}{2}$ since $c_{k+1} \geq 0$.

If we choose $\alpha_k \leq \frac{1}{L}$ for $k = 1, 2 \ldots, K$, then it follows from (5.50) that

$$\begin{aligned} & \mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \\ \leq & -\frac{\alpha_k(1-\beta_1)}{2}\left(\epsilon+\frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2\right] + (2-\beta_1)\frac{6\alpha_k^2DL\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 \\ & + \underbrace{\left(\frac{(2-\beta_1)^2}{(1-\beta_1)}\alpha_k\sigma^2 - b_k\right)}_{A^k}\mathbb{E}\Big[\sum_{i=1}^{p}\left((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\right)\Big] \\ & + \left(\frac{L}{2}+\rho_1+(1-\beta_1)^{-1}L\right)\mathbb{E}\left[\|\theta^{k+1}-\theta^k\|^2\right] \\ & + \sum_{d=1}^{D}\underbrace{\left((2-\beta_1)\epsilon^{-\frac{1}{2}}\left(\frac{L}{12}+\frac{c\alpha_k}{2d_{\max}}\right)+\rho_{d+1}-\rho_d\right)}_{B_d^k}\mathbb{E}\left[\|\theta^{k+1-d}-\theta^{k-d}\|^2\right]. \end{aligned} \tag{5.51}$$

To ensure $A^k \leq 0$ and $B_d^k \leq 0$, it is sufficient to choose $\{b_k\}$ and $\{\rho_d\}$ satisfying (with $\rho_{D+1} = 0$)

$$\frac{(2-\beta_1)^2}{(1-\beta_1)}\alpha_k\sigma^2 - b_k \leq 0, \quad k = 1, \cdots, K \tag{5.52}$$

$$(2-\beta_1)\epsilon^{-\frac{1}{2}}\left(\frac{L}{12}+\frac{c\alpha_k}{2d_{\max}}\right)+\rho_{d+1}-\rho_d \leq 0, \quad d = 1, \cdots, D. \tag{5.53}$$

Solve this system of linear equations and get

$$b_k = \frac{(2-\beta_1)^2}{(1-\beta_1)L}\sigma^2, \quad k = 1, \cdots, K \tag{5.54}$$

$$\rho_d = (2-\beta_1)\epsilon^{-\frac{1}{2}}\left(\frac{L}{12} + \frac{c}{2Ld_{\max}}\right)(D-d+1), \quad d = 1, \cdots, D \tag{5.55}$$

plugging which into (5.51) leads to the conclusion of Lemma 18.

5.5.5 Proof of Theorem 7

From the definition of $\mathcal{V}^k$, we have for any k, that

$$\begin{aligned}\mathbb{E}[\mathcal{V}^k] &\geq \mathcal{L}(\theta^k) - \mathcal{L}(\theta^*) - c_k\left\langle \nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\right\rangle + \sum_{d=1}^{D}\rho_d\|\theta^{k+1-d} - \theta^{k-d}\|^2 \\ &\geq -|c_k|\left\|\nabla\mathcal{L}(\theta^{k-1})\right\|\left\|(\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\right\| \\ &\geq -(1-\beta_1)^{-1}\alpha_k\sigma^2\epsilon^{-\frac{1}{2}} \end{aligned} \tag{5.56}$$

where we use Assumption 7 and Lemma 20.

By taking summation on (5.51) over $k = 0, \cdots, K-1$, it follows from that

$$\begin{aligned}&\frac{\alpha(1-\beta_1)}{2}\left(\epsilon + \frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2\right] \\ \leq& \frac{\mathbb{E}[\mathcal{V}^1] - \mathbb{E}[\mathcal{V}^{K+1}]}{K} + (2-\beta_1)\frac{6\alpha^2 DL\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 + \frac{(2-\beta_1)^2}{(1-\beta_1)}\sigma^2 pD\epsilon^{-\frac{1}{2}}\frac{\alpha}{K} \\ &+ \left(\frac{L}{2} + \rho_1 + (1-\beta_1)^{-1}L\right)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left[\|\theta^{k+1} - \theta^k\|^2\right] \\ \overset{(a)}{\leq}& \frac{\mathbb{E}[\mathcal{V}^1]}{K} + (2-\beta_1)\frac{6\alpha^2 DL\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 + (1-\beta_1)^{-1}\sigma^2\epsilon^{-\frac{1}{2}}\frac{\alpha}{K} + \frac{(2-\beta_1)^2}{(1-\beta_1)}\sigma^2 pD\epsilon^{-\frac{1}{2}}\frac{\alpha}{K} \\ &+ \left(\frac{L}{2} + \rho_1 + (1-\beta_1)^{-1}L\right)p(1-\beta_2)^{-1}(1-\beta_3)^{-1}\alpha^2 \end{aligned} \tag{5.57}$$

where (a) follows from (5.56) and Lemma 21.

Specifically, if we choose a constant stepsize $\alpha := \frac{\eta}{\sqrt{K}}$, where $\eta > 0$ is a constant, and define

$$\tilde{C}_1 := (2-\beta_1)6DL\epsilon^{-\frac{1}{2}} \tag{5.58}$$

and

$$\tilde{C}_2 := (1-\beta_1)^{-1}\epsilon^{-\frac{1}{2}} + \frac{(2-\beta_1)^2}{(1-\beta_1)} D\epsilon^{-\frac{1}{2}} \tag{5.59}$$

and

$$\tilde{C}_3 := \left(\frac{L}{2} + \rho_1 + (1-\beta_1)^{-1}L\right)(1-\beta_2)^{-1}(1-\beta_3)^{-1} \tag{5.60}$$

and

$$\tilde{C}_4 := \frac{1}{2}(1-\beta_1)\left(\epsilon + \frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}} \tag{5.61}$$

we can obtain from (5.57) that

$$\begin{aligned}
\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta^k)\|^2\right] \leq & \frac{\frac{\mathcal{L}(\theta^0)-\mathcal{L}(\theta^*)}{K} + \frac{\tilde{C}_1}{M}\sum_{m\in\mathcal{M}}\sigma_m^2\alpha^2 + \tilde{C}_2 p\sigma^2\frac{\alpha}{K} + \tilde{C}_3 p\alpha^2}{\alpha\tilde{C}_4} \\
\leq & \frac{\mathcal{L}(\theta^0)-\mathcal{L}(\theta^*)}{K\alpha\tilde{C}_4} + \frac{\tilde{C}_1\alpha}{\tilde{C}_4 M}\sum_{m\in\mathcal{M}}\sigma_m^2 + \tilde{C}_2 p\frac{\sigma^2}{K\tilde{C}_4} + \frac{\tilde{C}_3 p\alpha}{\tilde{C}_4} \\
= & \frac{(\mathcal{L}(\theta^0)-\mathcal{L}(\theta^*))C_4}{\sqrt{K}\eta} + \frac{C_1\eta}{\sqrt{K}M}\sum_{m\in\mathcal{M}}\sigma_m^2 + \frac{C_2 p\sigma^2}{K} + \frac{C_3 p\eta}{\sqrt{K}}
\end{aligned}$$

where we define $C_1 := \tilde{C}_1/\tilde{C}_4$, $C_2 := \tilde{C}_2/\tilde{C}_4$, $C_3 := \tilde{C}_3/\tilde{C}_4$, and $C_4 := 1/\tilde{C}_4$.

5.5.6 Proof of Theorem 8

By the PL-condition of $\mathcal{L}(\theta)$, we have

$$\begin{aligned}
& -\frac{\alpha_k(1-\beta_1)}{2}\left(\epsilon + \frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}}\mathbb{E}\left[\left\|\nabla\mathcal{L}(\theta^k)\right\|^2\right] \\
\leq & -\alpha_k\mu(1-\beta_1)\left(\epsilon + \frac{\sigma^2}{1-\beta_2}\right)^{-\frac{1}{2}}\mathbb{E}\left[\mathcal{L}(\theta^k) - \mathcal{L}(\theta^\star)\right] \\
\overset{(a)}{\leq} & -2\alpha_k\mu\tilde{C}_4\Big(\mathbb{E}[\mathcal{V}^k] + c_k\left\langle\nabla\mathcal{L}(\theta^{k-1}), (\epsilon I + \hat{V}^k)^{-\frac{1}{2}}h^k\right\rangle - b_k\sum_{d=0}^{D}\sum_{i=1}^{p}(\epsilon + \hat{v}_i^{k-d})^{-\frac{1}{2}} - \sum_{d=1}^{D}\rho_d\|\theta^{k+1-d} - \theta^{k-d}\|^2\Big) \\
\overset{(b)}{\leq} & -2\alpha_k\mu\tilde{C}_4\mathbb{E}[\mathcal{V}^k] + 2\alpha_k^2\mu\tilde{C}_4(1-\beta_1)^{-1}\sigma^2\epsilon^{-\frac{1}{2}} + 2\alpha_k\mu\tilde{C}_4 b_k\sum_{d=0}^{D}\sum_{i=1}^{p}\mathbb{E}\left[(\epsilon + \hat{v}_i^{k-d})^{-\frac{1}{2}}\right] \\
& + 2\alpha_k\mu\tilde{C}_4\sum_{d=1}^{D}\rho_d\mathbb{E}[\|\theta^{k+1-d} - \theta^{k-d}\|^2]
\end{aligned} \tag{5.62}$$

where (a) uses the definition of $\tilde{C}_4$ in (5.61), and (b) uses Assumption 7 and Lemma 20.

Plugging (5.62) into (5.50), we have

$$
\begin{aligned}
\mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \leq & -2\alpha_k\mu\tilde{C}_4\mathbb{E}[\mathcal{V}^k] + (2-\beta_1)\frac{6\alpha_k^2 DL\epsilon^{-\frac{1}{2}}}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 \\
& + \frac{(2-\beta_1)^2}{(1-\beta_1)}\alpha_k\sigma^2\mathbb{E}\Big[\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big] \\
& + b_{k+1}\sum_{i=1}^p\mathbb{E}\Big[(\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big] - (b_k - 2\alpha_k\mu\tilde{C}_4 b_k)\sum_{i=1}^p\mathbb{E}\Big[(\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}}\Big] \\
& + \sum_{d=1}^D(b_{k+1} - b_k + 2\alpha_k\mu\tilde{C}_4 b_k)\sum_{i=1}^p\mathbb{E}\Big[(\epsilon+\hat{v}_i^{k+1-d})^{-\frac{1}{2}}\Big] \\
& + \Big(\frac{L}{2} + \rho_1 + (1-\beta_1)^{-1}L\Big)p(1-\beta_2)^{-1}(1-\beta_3)^{-1}\alpha_k^2 + 2\alpha_k^2\mu\tilde{C}_4(1-\beta_1)^{-1}\sigma^2\epsilon^{-\frac{1}{2}} \\
& + \sum_{d=1}^D\Big((2-\beta_1)\epsilon^{-\frac{1}{2}}\Big(\frac{L}{12} + \frac{c\alpha_k}{2d_{\max}}\Big) + \rho_{d+1} - \rho_d + 2\alpha_k\mu\tilde{C}_4\rho_d\Big)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right].
\end{aligned}
\tag{5.63}
$$

If we choose b_k to ensure that $b_{k+1} \leq (1-2\alpha_k\mu\tilde{C}_4)b_k$, then we can obtain from (5.63) that

$$
\begin{aligned}
& \mathbb{E}[\mathcal{V}^{k+1}] - \mathbb{E}[\mathcal{V}^k] \\
\leq & -2\alpha_k\mu\tilde{C}_4\mathbb{E}[\mathcal{V}^k] + \frac{\tilde{C}_1}{M}\sum_{m\in\mathcal{M}}\sigma_m^2\alpha_k^2 + \tilde{C}_3 p\alpha_k^2 + 2\mu\tilde{C}_4(1-\beta_1)^{-1}\sigma^2\epsilon^{-\frac{1}{2}}\alpha_k^2 \\
& + \Big(\frac{(2-\beta_1)^2}{(1-\beta_1)}\alpha_k\sigma^2 - (1-2\alpha_k\mu\tilde{C}_4)b_k\Big)\mathbb{E}\Big[\sum_{i=1}^p\Big((\epsilon+\hat{v}_i^{k-D})^{-\frac{1}{2}} - (\epsilon+\hat{v}_i^{k+1})^{-\frac{1}{2}}\Big)\Big] \\
& + \sum_{d=1}^D\Big((2-\beta_1)\epsilon^{-\frac{1}{2}}\Big(\frac{L}{12} + \frac{c\alpha_k}{2d_{\max}}\Big) + \rho_{d+1} - \rho_d + 2\alpha_k\mu\tilde{C}_4\rho_d\Big)\mathbb{E}\left[\|\theta^{k+1-d} - \theta^{k-d}\|^2\right].
\end{aligned}
\tag{5.64}
$$

If $\alpha_k \leq \frac{1}{L}$, we choose parameters $\{b_k, \rho_d\}$ to guarantee that

$$
\frac{(2-\beta_1)^2}{(1-\beta_1)L}\sigma^2 - \Big(1 - \frac{2\mu\tilde{C}_4}{L}\Big)b_k \leq 0, \quad \forall k \tag{5.65}
$$

$$
(2-\beta_1)\Big(\frac{L}{12} + \frac{c}{2Ld_{\max}}\Big)\epsilon^{-\frac{1}{2}} + \rho_{d+1} - \Big(1 - \frac{2\mu\tilde{C}_4}{L}\Big)\rho_d \leq 0, \quad d = 1, \cdots, D \tag{5.66}
$$

and choose $\beta_1, \beta_2, \epsilon$ to ensure that $1 - \frac{2\mu\tilde{C}_4}{L} \geq 0$.

Then we have

$$\mathbb{E}[\mathcal{V}^{k+1}] \leq \left(1-2\alpha_k\mu\tilde{C}_4\right)\mathbb{E}[\mathcal{V}^k] + \Bigg(\underbrace{\frac{\tilde{C}_1}{M}\sum_{m\in\mathcal{M}}\sigma_m^2 + \tilde{C}_3 p + 2\mu\tilde{C}_4(1-\beta_1)^{-1}\sigma^2\epsilon^{-\frac{1}{2}}}_{\tilde{C}_5}\Bigg)\alpha_k^2$$

$$\leq \prod_{j=0}^{k}(1-2\alpha_j\mu\tilde{C}_4)\mathbb{E}[\mathcal{V}^0] + \sum_{j=0}^{k}\alpha_j^2\prod_{i=j+1}^{k}(1-2\alpha_i\mu\tilde{C}_4)\tilde{C}_5. \tag{5.67}$$

If we choose $\alpha_k = \frac{1}{\mu(k+K_0)\tilde{C}_4} \leq \frac{1}{L}$, where K_0 is a sufficiently large constant to ensure that α_k satisfies the aforementioned conditions, then we have

$$\begin{aligned}
\mathbb{E}[\mathcal{V}^K] \leq & \mathbb{E}[\mathcal{V}^0]\prod_{k=0}^{K-1}(1-2\alpha_k\mu\tilde{C}_4) + \tilde{C}_5\sum_{k=0}^{K-1}\alpha_k^2\prod_{j=k+1}^{K-1}(1-2\alpha_j\mu\tilde{C}_4) \\
\leq & \mathbb{E}[\mathcal{V}^0]\prod_{k=0}^{K-1}\frac{k+K_0-2}{k+K_0} + \frac{\tilde{C}_5}{\mu^2\tilde{C}_4^2}\sum_{k=0}^{K-1}\frac{1}{(k+K_0)^2}\prod_{j=k+1}^{K-1}\frac{j+K_0-2}{j+K_0} \\
\leq & \frac{(K_0-2)(K_0-1)}{(K+K_0-2)(K+K_0-1)}\mathbb{E}[\mathcal{V}^0] + \frac{\tilde{C}_5}{\mu^2\tilde{C}_4^2}\sum_{k=0}^{K-1}\frac{(k+K_0-1)}{(k+K_0)(K+K_0-2)(K+K_0-2)} \\
\leq & \frac{(K_0-1)^2}{(K+K_0-1)^2}\mathbb{E}[\mathcal{V}^0] + \frac{\tilde{C}_5 K}{\mu^2\tilde{C}_4^2(K+K_0-1)^2} \\
= & \frac{(K_0-1)^2}{(K+K_0-1)^2}(\mathcal{L}(\theta^0)-\mathcal{L}(\theta^\star)) + \frac{\tilde{C}_5 K}{\mu^2\tilde{C}_4^2(K+K_0-2)^2}
\end{aligned}$$

from which the proof is complete.

CHAPTER 6

Summary

In this final chapter, we provide a summary of the main results discussed in this thesis.

6.1 Thesis summary

This thesis focuses on developing new stochastic optimization methods to tackle two fundamental classes of machine learning problems: C1) stochastic nested problems, where one subproblem builds upon the solution of others; and, C2) stochastic distributed problems, where the subproblems are coupled through sharing data and/or variables.

In the first part of the thesis, which contains Chapters 2 and 3, the aim was to develop sample-efficient stochastic optimization methods amenable to solve stochastic nested problems in C1. The key take-home message there is that for a class of stochastic nested problems, our single-loop stochastic optimization methods can achieve the same sample complexity as the stochastic gradient descent method for classic problems without stochastic nested structures.

In Chapter 2, we introduced a new method termed SCSC for solving the class of stochastic compositional optimization problems. SCSC runs in a single-time scale with a single loop, uses a fixed batch size. Remarkably, it converges at the same rate as the SGD method for non-compositional stochastic optimization. This is achieved by making a careful improvement to a popular stochastic compositional gradient method.

In Chapter 3, we developed a new stochastic gradient estimator for bilevel optimization problems. When running SGD on top of this stochastic bilevel gradient, the resultant

STABLE algorithm runs in a single loop fashion, and uses a single-timescale update. To achieve an ϵ-stationary point in the nonconvex case, STABLE requires $\mathcal{O}(\epsilon^{-2})$ samples, and to achieve an ϵ-optimal solution in the strongly-convex case, STABLE requires $\mathcal{O}(\epsilon^{-1})$ samples. In both cases, STABLE matches the sample complexity of SGD for single-level problems.

In the second part of the thesis, which contains Chapters 4 and 5, the aim was to develop communication-efficient distributed stochastic optimization methods amenable to solve stochastic distributed problems in C2. The key take-home message there is that by exploiting the gradient innovations, our new distributed stochastic optimization methods can achieve the same convergence rate but save significantly communication overhead.

In Chapter 4, we developed a class of communication-efficient variants of SGD that we term LASG. The LASG methods leverage a set of adaptive communication rules to detect and then skip less informative or redundant communication rounds between the server and workers during distributed learning. To further reduce communication bandwidth, the quantized version of LASG is also presented. Both LASG and their quantized version are simple to implement, and have convergence rate comparable to the original SGD.

In Chapter 5, we have developed a communication-adaptive distributed Adam method that we term CADA, which endows an additional dimension of adaptivity to Adam tailored for its distributed implementation. CADA method leverages a set of adaptive communication rules to detect and then skip less informative communication rounds between the server and workers during distributed learning. All CADA variants are simple to implement, and have convergence rate comparable to the original Adam.

6.2 Future research directions

• **Distributed stochastic nested optimization.** With our promising results of stochastic nested optimization in this thesis, we plan to tackle the decentralized formulation of (1.1). We plan to answer the following questions: Whether the decentralized optimization algorithms for (1.1) can achieve the same order of sample complexity obtained by decentralized SGD for the single-level problems? If the answer is yes, what are the complexities of it? Building upon prior work on decentralized optimization for single-level problems, we will conduct a thorough analysis in terms of the iteration, sample and communication complexities for the decentralized stochastic bilevel and compositional algorithms.

- **Accelerated stochastic nested optimization methods.** Going beyond the SGD-based stochastic nested optimization methods, we are motivated to develop accelerated stochastic compositional methods that incorporate the momentum and acceleration techniques [86, 80, 31] into the update of the outer variable θ.

Starting again from the two-layer compositional optimization case, we plan to develop accelerated compositional gradient methods relying on the following ODE

$$\ddot{\theta}(t) = -2\sqrt{\mu}\dot{\theta}(t) - \alpha\nabla g_2(\theta(t))\nabla g_1(y(t)) \tag{6.1}$$

where $\mu > 0$ is the strong convexity constant of $F(\theta)$. This is non-trivial considering their non-compositional counterparts because the momentum update of θ is intertwined with the tracking variables for the inner functions. Once we develop the accelerated methods in the strongly convex case, we will then extend it to the convex and nonconvex cases.

REFERENCES

[1] Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. In *Proc. Conf. Empirical Methods Natural Language Process.*, pages 440–445, Copenhagen, Denmark, Sep 2017.

[2] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. In *Proc. Advances in Neural Info. Process. Syst.*, pages 1709–1720, Long Beach, CA, Dec 2017.

[3] Dan Alistarh, Torsten Hoefler, Mikael Johansson, Nikola Konstantinov, Sarit Khirirat, and Cédric Renggli. The convergence of sparsified gradient methods. In *Proc. Advances in Neural Info. Process. Syst.*, pages 5973–5983, Montreal, Canada, Dec 2018.

[4] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. SignSGD: Compressed optimisation for non-convex problems. In *Proc. Intl. Conf. Machine Learn.*, pages 559–568, Stockholm, Sweden, Jul 2018.

[5] Jose Blanchet, Donald Goldfarb, Garud Iyengar, Fengpei Li, and Chaoxu Zhou. Unbiased simulation for optimizing stochastic function compositions. *arXiv preprint:1711.07564*, November 2017.

[6] Zalán Borsos, Mojmír Mutnỳ, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. In *Proc. Advances in Neural Info. Process. Syst.*, Virtual, December 2020.

[7] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[8] Tianyi Chen, Georgios Giannakis, Tao Sun, and Wotao Yin. LAG: Lazily aggregated gradient for communication-efficient distributed learning. In *Proc. Advances in Neural Info. Process. Syst.*, pages 5050–5060, Montreal, Canada, Dec 2018.

[9] Tianyi Chen, Ziye Guo, Yuejiao Sun, and Wotao Yin. Cada: Communication-adaptive distributed adam. In *International Conference on Artificial Intelligence and Statistics*, pages 613–621. PMLR, 2021.

[10] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Lasg: Lazily aggregated stochastic gradients for communication-efficient distributed learning. *arXiv preprint arXiv:2002.11360*, 2020.

[11] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. *arXiv preprint:2008.10847*, August 2020.

[12] Tianyi Chen, Yuejiao Sun, and Wotao Yin. A single-timescale stochastic bilevel optimization method. *arXiv preprint arXiv:2102.04671*, 2021.

[13] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of Adam-type algorithms for non-convex optimization. In *Proc. Intl. Conf. Learn. Representations*, New Orleans, LA, May 2019.

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

[15] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. *Proc. Advances in Neural Info. Process. Syst.*, 32, December 2019.

[16] Christoph Dann, Gerhard Neumann, Jan Peters, et al. Policy evaluation with temporal differences: A survey and comparison. *J. Machine Learning Res.*, 15:809–883, 2014.

[17] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. Large scale distributed deep networks. In *Proc. Advances in Neural Info. Process. Syst.*, pages 1223–1231, Lake Tahoe, NV, 2012.

[18] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Proc. Advances in Neural Info. Process. Syst.*, pages 1646–1654, Montreal, Canada, December 2014.

[19] Alexandre Défossez, Léon Bottou, Francis Bach, and Nicolas Usunier. On the convergence of Adam and Adagrad. *arXiv preprint:2003.02395*, March 2020.

[20] Stephan Dempe and Alain Zemkoho. *Bilevel Optimization.* Springer, 2020.

[21] Adithya M Devraj and Jianshu Chen. Stochastic variance reduced primal dual algorithms for empirical composition optimization. In *Proc. Advances in Neural Info. Process. Syst.*, pages 9878–9888, Vancouver, Canada, December 2019.

[22] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Machine Learning Res.*, 12(Jul):2121–2159, 2011.

[23] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. *arXiv preprint:1908.10400*, August 2019.

[24] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Provably convergent policy gradient methods for model-agnostic meta-reinforcement learning. *arXiv preprint:2002.05135*, February 2020.

[25] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. SPIDER: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Proc. Advances in Neural Info. Process. Syst.*, pages 689–699, Montreal, Canada, December 2018.

[26] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. Intl. Conf. Machine Learn.*, pages 1126–1135, Sydney, Australia, June 2017.

[27] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In *Proc. Intl. Conf. Machine Learn.*, pages 1920–1930, Long Beach, CA, June 2019.

[28] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *Proc. Intl. Conf. Machine Learn.*, pages 1568–1577, Vienna, Austria, June 2018.

[29] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[30] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[31] Saeed Ghadimi and Guanghui Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016.

[32] Saeed Ghadimi, Andrzej Ruszczynski, and Mengdi Wang. A single timescale stochastic approximation method for nested stochastic optimization. *SIAM Journal on Optimization*, 30(1):960–979, March 2020.

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

[34] G B Giannakis, Q Ling, G Mateos, I D Schizas, and H Zhu. Decentralized Learning for Wireless Communications and Networking. In *Splitting Methods in Communication and Imaging, Science and Engineering*. Springer, New York, 2016.

[35] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *Proc. Intl. Conf. Machine Learn.*, pages 3748–3758, virtual, July 2020.

[36] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck Cadambe. Local sgd with periodic averaging: Tighter analysis and adaptive synchronization. In *Proc. Conf. Neural Info. Process. Syst.*, pages 11080–11092, Vancouver, Canada, December 2019.

[37] M. Hong, H.-T. Wai, Z. Wang, and Z. Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint:2007.05170*, 2020.

[38] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Sebastian Stich, and Peter Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint:1904.05115*, April 2019.

[39] Wenqing Hu, Chris Junchi Li, Xiangru Lian, Ji Liu, and Huizhuo Yuan. Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent. In *Proc. Advances in Neural Info. Process. Syst.*, pages 6926–6935, Vancouver, Canada, December 2019.

[40] Zhouyuan Huo, Bin Gu, Ji Liu, and Heng Huang. Accelerated method for stochastic composition optimization with nonsmooth regularization. In *Proc. of Assoc. for Advanc. Artif. Intell.*, New Orleans, LA, February 2018.

[41] Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Ion Stoica, Raman Arora, et al. Communication-efficient distributed SGD with sketching. In *Proc. Conf. Neural Info. Process. Syst.*, pages 13144–13154, Vancouver, Canada, December 2019.

[42] Martin Jaggi, Virginia Smith, Martin Takác, Jonathan Terhorst, Sanjay Krishnan, Thomas Hofmann, and Michael I Jordan. Communication-efficient distributed dual coordinate ascent. In *Proc. Advances in Neural Info. Process. Syst.*, pages 3068–3076, Montreal, Canada, December 2014.

[43] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Multi-step model-agnostic meta-learning: Convergence and improved algorithms. *arXiv preprint:2002.07836*, February 2020.

[44] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Provably faster algorithms for bilevel optimization and applications to meta-learning. *arXiv preprint:2010.07962*, 2020.

[45] Peng Jiang and Gagan Agrawal. A linear speedup analysis of distributed deep learning with sparse and quantized communication. In *Proc. Conf. Neural Info. Process. Syst.*, pages 2525–2536, Montreal, Canada, Dec 2018.

[46] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Proc. Advances in Neural Info. Process. Syst.*, pages 315–323, Lake Tahoe, NV, December 2013.

[47] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Proc. Conf. Neural Info. Process. Syst.*, pages 315–323, 2013.

[48] Michael I Jordan, Jason D Lee, and Yun Yang. Communication-efficient distributed statistical inference. *J. American Statistical Association*, to appear, 2018.

[49] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint:1912.04977*, December 2019.

[50] Michael Kamp, Linara Adilova, Joachim Sicking, Fabian Hüger, Peter Schlicht, Tim Wirtz, and Stefan Wrobel. Efficient decentralized deep learning by dynamic model averaging. In *Euro. Conf. Machine Learn. Knowledge Disc. Data.,*, pages 393–409, Dublin, Ireland, 2018.

[51] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Proc. Euro. Conf. Machine Learn.*, pages 795–811, Riva del Garda, Italy, September 2016.

[52] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for on-device federated learning. In *Proc. Intl. Conf. Machine Learn.*, July 2020.

[53] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *Proc. Intl. Conf. Machine Learn.*, pages 3252–3261, Long Beach, CA, June 2019.

[54] M Khodak, M Balcan, and A Talwalkar. Provable guarantees for gradient-based meta-learning. In *Proc. Intl. Conf. Machine Learn.*, Long Beach, CA, June 2019.

[55] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint:1412.6980*, December 2014.

[56] V. Konda and V. Borkar. Actor-critic-type learning algorithms for markov decision processes. *SIAM Journal on Control and Optimization*, 38(1):94–123, 1999.

[57] Gautam Kunapuli, Kristin P Bennett, Jing Hu, and Jong-Shi Pang. Classification model selection via bilevel programming. *Optimization Methods & Software*, 23(4):475–489, 2008.

[58] Karl Kunisch and Thomas Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.

[59] Boyue Li, Shicong Cen, Yuxin Chen, and Yuejie Chi. Communication-efficient distributed optimization in networks with gradient tracking and variance reduction. In *Proc. Intl. Conf. on Artif. Intell. and Stat.*, pages 1662–1672, Palermo, Italy, June 2020.

[60] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

[61] Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *Proc. Intl. Conf. on Artif. Intell. and Stat.*, pages 983–992, Okinawa, Japan, April 2019.

[62] Xiangru Lian, Mengdi Wang, and Ji Liu. Finite-sum composition optimization via variance reduced gradient descent. In *Proc. Intl. Conf. on Artif. Intell. and Stat.*, Fort Lauderdale, FL, April 2017.

[63] Xiangru Lian, Huan Zhang, Cho-Jui Hsieh, Yijun Huang, and Ji Liu. A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order. In *Proc. Conf. Neural Info. Process. Syst.*, pages 3054–3062, Barcelona, Spain, December 2016.

[64] Tao Lin, Sebastian U Stich, Kumar Kshitij Patel, and Martin Jaggi. Don't use large mini-batches, use local sgd. In *Proc. Intl. Conf. Learn. Representations*, Addis Ababa, Ethiopia, April 2020.

[65] Tianyi Lin, Chenyou Fan, Mengdi Wang, and Michael I Jordan. Improved oracle complexity for stochastic compositional variance reduced gradient. *arXiv preprint:1806.00458*, June 2018.

[66] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and William J Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *Proc. Intl. Conf. Learn. Representations*, Vancouver, Canada, Apr 2018.

[67] Hao Liu, Richard Socher, and Caiming Xiong. Taming maml: Efficient unbiased meta-reinforcement learning. In *Proc. Intl. Conf. Machine Learn.*, pages 4061–4071, Long Beach, CA, June 2019.

[68] Risheng Liu, Pan Mu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *Proc. of International Conference on Machine Learning*, pages 6305–6315, Virtual, July 2020.

[69] Yaohua Liu, Wei Xu, Gang Wu, Zhi Tian, and Qing Ling. Communication-censored ADMM for decentralized consensus optimization. *IEEE Trans. Sig. Proc.*, 67(10):2565–2579, March 2019.

[70] Chenxin Ma, Jakub Konečnỳ, Martin Jaggi, Virginia Smith, Michael I Jordan, Peter Richtárik, and Martin Takáč. Distributed optimization with arbitrary local solvers. *Optimization Methods and Software*, 32(4):813–848, July 2017.

[71] Xianghui Mao, Kun Yuan, Yubin Hu, Yuantao Gu, Ali H Sayed, and Wotao Yin. Walkman: A communication-efficient random-walk algorithm for decentralized optimization. *IEEE Trans. Sig. Proc.*, 68:2513–2528, March 2020.

[72] B McMahan and Daniel Ramage. Federated learning: Collaborative machine learning without centralized training data. *Google Research Blog*, April 2017.

[73] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proc. Intl. Conf. Artificial Intell. and Stat.*, pages 1273–1282, Fort Lauderdale, FL, April 2017.

[74] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proc. Intl. Conf. on Artif. Intell. and Stat.*, pages 1273–1282, Fort Lauderdale, Florida, Apr 2017.

[75] Eric J Msechu and Georgios B Giannakis. Sensor-centric data reduction for estimation with WSNs via censoring and quantization. *IEEE Trans. Sig. Proc.*, 60(1):400–414, Jan 2011.

[76] Angelia Nedić, Alex Olshevsky, and Michael Rabbat. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106(5):953–976, May 2018.

[77] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Trans. Automat. Control*, 54(1):48–61, January 2009.

[78] Arkadiĭ Semenovich Nemirovsky and David Borisovich Yudin. *Problem Complexity And Method Efficiency In Optimization.* Wiley Interscience Series in Discrete Mathematics, 1983.

[79] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A basic course*, volume 87. Springer, Berlin, Germany, 2013.

[80] Yurii E Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. In *Doklady AN USSR*, volume 269, pages 543–547, 1983.

[81] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *Proc. Intl. Conf. Machine Learn.*, pages 2613–2621, Sydney, Australia, August 2017.

[82] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint:1803.02999*, March 2018.

[83] Jorge Nocedal and Stephen Wright. *Numerical Optimization.* Springer, Berlin, Germany, 2006.

[84] Jihong Park, Sumudu Samarakoon, Mehdi Bennis, and Mérouane Debbah. Wireless network intelligence at the edge. *Proc. of the IEEE*, 107(11):2204–2239, November 2019.

[85] Larry L Peterson and Bruce S Davie. *Computer Networks: A Systems Approach.* Morgan Kaufman, Burlington, MA, 2007.

[86] Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[87] Michael G Rabbat and Robert D Nowak. Quantized incremental algorithms for distributed optimization. *IEEE J. Sel. Areas Commun.*, 23(4):798–808, April 2005.

[88] Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. In *Proc. Advances in Neural Info. Process. Syst.*, pages 113–124, Vancouver, Canada, December 2019.

[89] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. *arXiv preprint arXiv:1109.5647*, 2011.

[90] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *Proc. Intl. Conf. Learn. Representations*, Toulon, France, May 2017.

[91] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint:2003.00295*, March 2020.

[92] Amirhossein Reisizadeh, Hossein Taheri, Aryan Mokhtari, Hamed Hassani, and Ramtin Pedarsani. Robust and communication-efficient collaborative learning. In *Proc. Conf. Neural Info. Process. Syst.*, pages 8386–8397, Vancouver, Canada, December 2019.

[93] H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22(3):400–407, September 1951.

[94] Andrzej Ruszczynski. A stochastic subgradient method for nonsmooth nonconvex multi-level composition optimization. *arXiv preprint:2001.10669*, January 2020.

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

[96] Anit Kumar Sahu, Dusan Jakovetic, Dragana Bajovic, and Soummya Kar. Communication-efficient distributed strongly convex stochastic optimization: Non-asymptotic rates. *arXiv preprint:1809.02920*, September 2018.

[97] J REDDI Sashank, KALE Satyen, and KUMAR Sanjiv. On the convergence of Adam and beyond. In *Proc. Intl. Conf. Learn. Representations*, Vancouver, Canada, April 2018.

[98] Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns. In *Proc. Conf. Intl. Speech Comm. Assoc.*, Singapore, Sept 2014.

[99] Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. Truncated back-propagation for bilevel optimization. In *Proc. Intl. Conf. on Artif. Intell. and Stat.*, pages 1723–1732, Naha, Okinawa, Japan, April 2019.

[100] Ohad Shamir, Nati Srebro, and Tong Zhang. Communication-efficient distributed optimization using an approximate newton-type method. In *Proc. Intl. Conf. Machine Learn.*, pages 1000–1008, Beijing, China, Jun 2014.

[101] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, Philadelphia, PA, 2009.

[102] Osvaldo Simeone, Sangwoo Park, and Joonhyuk Kang. From learning to meta-learning: Reduced training overhead and complexity for communication systems. *arXiv preprint:2001.01227*, January 2020.

[103] Xingyou Song, Wenbo Gao, Yuxiang Yang, Krzysztof Choromanski, Aldo Pacchiano, and Yunhao Tang. ES-MAML: Simple hessian-free meta learning. In *Proc. Intl. Conf. Learn. Representations*, Addis Ababa, Ethiopia, April 2020.

[104] H. Van Stackelberg. *The Theory of Market Economy.* Oxford University Press, 1952.

[105] Sebastian U. Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified SGD with memory. In *Proc. Advances in Neural Info. Process. Syst.*, pages 4447–4458, Montreal, Canada, Dec 2018.

[106] Sebastian Urban Stich. Local sgd converges fast and communicates little. In *Proc. Intl. Conf. Learn. Representations*, New Orleans, LA, May 2019.

[107] Nikko Strom. Scalable distributed DNN training using commodity gpu cloud computing. In *Proc. Conf. Intl. Speech Comm. Assoc.*, Dresden, Germany, September 2015.

[108] Haoran Sun, Songtao Lu, and Mingyi Hong. Improving the sample and communication complexity for decentralized non-convex optimization: A joint gradient estimation and tracking approach. *arXiv preprint:1910.05857*, 2019.

[109] Jun Sun, Tianyi Chen, Georgios Giannakis, and Zaiyue Yang. Communication-efficient distributed learning via lazily aggregated quantized gradients. In *Proc. Conf. Neural Info. Process. Syst.*, page to appear, Vancouver, Canada, Dec 2019.

[110] Hanlin Tang, Shaoduo Gan, Ce Zhang, Tong Zhang, and Ji Liu. Communication compression for decentralized training. In *Proc. Conf. Neural Info. Process. Syst.*, pages 7652–7662, Montreal, Canada, December 2018.

[111] Rasul Tutunov, Minne Li, Jun Wang, and Haitham Bou-Ammar. Compositional Adam: An adaptive compositional solver. *arXiv preprint:2002.03755*, February 2020.

[112] Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. PowerSGD: Practical low-rank gradient compression for distributed optimization. In *Proc. Conf. Neural Info. Process. Syst.*, pages 14236–14245, Vancouver, Canada, December 2019.

[113] Guanghui Wang, Shiyin Lu, Weiwei Tu, and Lijun Zhang. SAdam: A variant of adam for strongly convex functions. In *Proc. Intl. Conf. Learn. Representations*, 2020.

[114] Hongyi Wang, Scott Sievert, Shengchao Liu, Zachary Charles, Dimitris Papailiopoulos, and Stephen Wright. Atomo: Communication-efficient learning via atomic sparsification. In *Proc. Conf. Neural Info. Process. Syst.*, pages 9850–9861, Montreal, Canada, Dec 2018.

[115] Jianyu Wang and Gauri Joshi. Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms. *arXiv preprint:1808.07576*, August 2018.

[116] Jianyu Wang, Vinayak Tantia, Nicolas Ballas, and Michael Rabbat. SlowMo: Improving communication-efficient distributed SGD with slow momentum. In *Proc. Intl. Conf. Learn. Representations*, 2020.

[117] Mengdi Wang, Ethan X Fang, and Han Liu. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*, 161(1-2):419–449, January 2017.

[118] Mengdi Wang, Ji Liu, and Ethan Fang. Accelerating stochastic composition optimization. *J. Machine Learning Res.*, 18(1):3721–3743, 2017.

[119] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. Gradient sparsification for communication-efficient distributed optimization. In *Proc. Conf. Neural Info. Process. Syst.*, pages 1299–1309, Montreal, Canada, Dec 2018.

[120] Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. In *Proc. Intl. Conf. Machine Learn.*, pages 6677–6686, Long Beach, CA, June 2019.

[121] Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Proc. Conf. Neural Info. Process. Syst.*, pages 1509–1519, Long Beach, CA, Dec 2017.

[122] Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized sgd and its applications to large-scale distributed optimization. In *Proc. Intl. Conf. Machine Learn.*, pages 5325–5333, Stockholm, Sweden, Jul 2018.

[123] Yangyang Xu, Colin Sutcher-Shepard, Yibo Xu, and Jie Chen. Asynchronous parallel adaptive stochastic gradient methods. *arXiv preprint:2002.09095*, February 2020.

[124] Yibo Xu and Yangyang Xu. Katyusha acceleration for convex finite-sum compositional optimization. *arXiv preprint:1910.11217*, October 2019.

[125] Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. In *Proc. Intl. Conf. Machine Learn.*, pages 7184–7193, Long Beach, CA, June 2019.

[126] Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proc. of Assoc. for Advanc. Artif. Intell.*, volume 33, pages 5693–5700, 2019.

[127] Yue Yu and Longbo Huang. Fast stochastic variance reduced ADMM for stochastic composition optimization. In *Proc. Intl. Joint Conf. Artif. Intell.*, pages 3364–3370, Melbourne, Australia, August 2017.

[128] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint:1212.5701*, December 2012.

[129] Hantian Zhang, Jerry Li, Kaan Kara, Dan Alistarh, Ji Liu, and Ce Zhang. Zipml: Training linear models with end-to-end low precision, and a little bit of deep learning. In *Proc. Intl. Conf. Machine Learn.*, pages 4035–4043, Sydney, Australia, Aug 2017.

[130] Junyu Zhang and Lin Xiao. A composite randomized incremental gradient method. In *Proc. Intl. Conf. Machine Learn.*, pages 7454–7462, Long Beach, CA, June 2019.

[131] Junyu Zhang and Lin Xiao. A stochastic composite gradient method with incremental variance reduction. In *Proc. Advances in Neural Info. Process. Syst.*, pages 9075–9085, Vancouver, Canada, December 2019.

[132] Sixin Zhang, Anna E Choromanska, and Yann LeCun. Deep learning with elastic averaging SGD. In *Proc. Conf. Neural Info. Process. Syst.*, pages 685–693, Montreal, Canada, Dec 2015.

[133] Yuchen Zhang and Xiao Lin. DiSCO: Distributed optimization for self-concordant empirical loss. In *Proc. Intl. Conf. Machine Learn.*, pages 362–370, Lille, France, June 2015.

[134] Pan Zhou, Xiaotong Yuan, Huan Xu, Shuicheng Yan, and Jiashi Feng. Efficient meta learning via minibatch proximal update. In *Proc. Advances in Neural Info. Process. Syst.*, pages 1532–1542, Vancouver, Canada, December 2019.